LEVELT

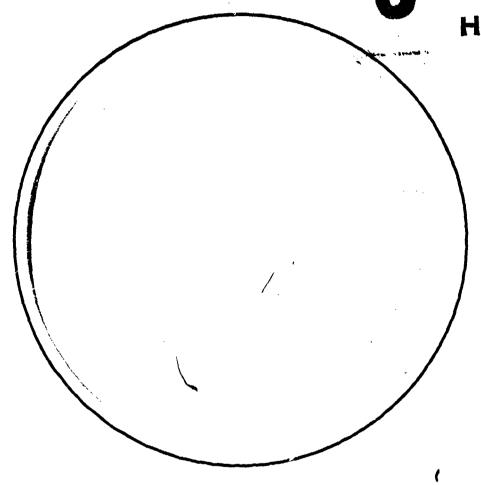




DTIC ELECTE DEC 3 1 1981

AD A 109134

TE FILE COPY



# Center for Information Systems Research

Massachusetts Institute of Technology Alfred P. Sloan School of Management 50 Memorial Drive Cambridge, Massachusetts, 02139 617 253-1000

8112 31103

SECURITY CLASSIFICATION OF THIS PAGE (When Date Entered)

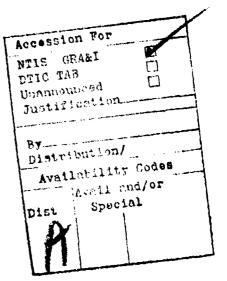
REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM			
Technical Report #6 2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER			
4. TITLE (and Subtitio)	3. TYPE OF REPORT & PERIOD COVERED			
A Study of the Multicache-Consistency Problem in Multi-Processor Computer Systems	6. PERFORMING ORG. REPORT NUMBER			
7. AUTHOR(s)	M010-8109-06 TIK-6			
Tarek K. Abdel-Hamid Stuart E. Madnick	N00039-78-G-0160			
PERFORMING ORGANIZATION NAME AND ADDRESS Slozn School of Management Massachusetts Institute of Technology Cambridge, Massachusetts 02139	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT HUMBERS			
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE September 1981			
	13. NUMBER OF PAGES			
14. MONITORING AGENCY NAME & ADDRESS(II different from Controlling Office)	Unclassified			
	15. DECLASSIFICATION/DOWNGRADING			
Approved for public release; distribution unlimited  17. DISTRIBUTION STATEMENT (of the electroct entered in Block 20, 11 different from Report)				
18. SUPPLEMENTARY NOTES				
·				
19. KEY WORDS (Gontinue on reverse elde if necessary and identify by block number)  Bus, Cache, Multicache-consistency, Multiprocessors, Performance Evaluation, Simulation				
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)				
The nature of the consistency problem in mu briefly discussed, together with an explanation approaches proposed in the literature to handle problem, called the "Common-Cache / Pended Trans	of the three common it. A new solution to the			
DD 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE 38	7679 \			

SECURITY CLASSIFICATION OF THIS PAGE (When Date Brief

#### 20. Abstract

approach, is developed and discussed. The "CC/PTB" approach attempts to minimize performance degradation by eliminating the overhead of maintaining cache-consistency. Its two distinctive features are: Firstly, the conventional private cache per processor organization is replaced by one where a pool of cache-modules is commonly shared by all processors. Secondly, the Pended Transaction Bus (PTB) is used as the interconnection protocol that connects the processors and the cache-modules.

The performance of the CC/PTB approach is evaluated using a highly detailed simulation model with favorable results.



#### **ABSTRACT**

This paper is a report on an ongoing research project at the M.I.T. Sloan School of Management to study the multicache-consistency problem in multi-processor computer systems.

The nature of the consistency problem in multicache memory systems is briefly discussed, together with an explanation of the three common approaches proposed in the literature to handle it. A new solution to the problem, called the "Common-Cache / Pended Transaction Bus" (CC/PTB) approach, is developed and discussed. The "CC/PTB" approach attempts to minimize performance degradation by eliminating the overhead of maintaining cache-consistency. Its two distinctive features are: Firstly, the conventional private cache per processor organization is replaced by one where a pool of cache-modules is commonly shared by all processors. Secondly, the Pended Transaction/Bus (PTB) is used as the interconnection protocol that connects the processors and the cache-modules.

The performance of the CC/PTB approach is evaluated using a highly detailed simulation model with favorable results.

# TABLE OF CONTENTS

I. T	THE MULTICACHE-CONSISTENCY PROBLEM: AN INTRODUCTION1
II. I	INFOPLEX: A MULTI-PROCESSOR COMPUTER SYSTEM
C I I	THREE COMMON APPROACHES FOR SOLVING THE CACHE-CONSISTENCY PROBLEM
I	THE "COMMON-CACHE/PENDED TRANSACTION BUS" APPROACH25 IV.1 Introduction
V V V V	EVALUATING THE PERFORMANCE OF THE  COMMON-CACHE/FENDED TRANSACTION BUS" APPROACH
VI.	CONCLUSION59
вівшо	OGRAPHY61
APPENI	DIX (I): The "OPT" Program

#### I. THE MULTICACHE-CONSISTENCY PROBLEM: AN INTRODUCTION

A "consistency problem" refers, in general, to a situation where two or more entries representing the same "fact" in a data base differ (i.e., are inconsistent). This, of course, can only occur when redundancy exists. In this paper we will be concerned with the "consistency problem" that arises in cache-based memory systems.

A cache memory system (Kaplan and Winder, 1973) represents a type of memory hierarchy that thempts to bridge the CPU-main memory speed gap by the use of a small, high speed random access memory whose cost per bit is higher than that of main memory, but whose total cost is relatively small because of the small size.

Conceptually, this configuration has analogies with paging systems (Matick, 1977). The implementations, however, are far apart because of speed considerations. In contrast to a puging system, a cache is managed by hardware algorithms, provides a smaller ratio of memory access times (e.g., 10:1 rather than 1000:1), and deals with smaller

blocks of data (64 bytes for example rather than 40%).

In a cache system, all data are referenced by their main memory address. At any given time, a certain subset of the contents of main memory is contained in the cache level. If a processor then requests a data item in this subset, the request is serviced at the cache level.

A cache-based system works "well" for two basic reasons:

First, executing programs tend to re-use instructions and data; and second, programs tend to use instructions and data near recently used instructions and data. The first property means that once information is fetched from main memory to cache, subsequent accesses to it are at cache speed. The second property means that if a request to main memory is satisfied by bringing into a cache a block of information larger than is immediately needed, the additional information is likely to be needed soon, and its presence in the cache will save one or more references to main memory.

In this paper we will refer to the block of information that constitutes the minimum amount of data which may be transmitted between the cache and main memory and which is also the allocation unit in the cache as the "cache line." All bytes of a cache line are, therefore, simultaneously all present or all absent from the cache. A directory is usually used to record the main memory acciresses of all lines in the cache.

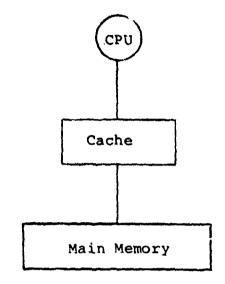
It is easy to demonstrate how inconsistency can develop in

cache-based systems due to the existence of redundancy. Consider first the simple case of a single-cache organization (Figure 1(a)).

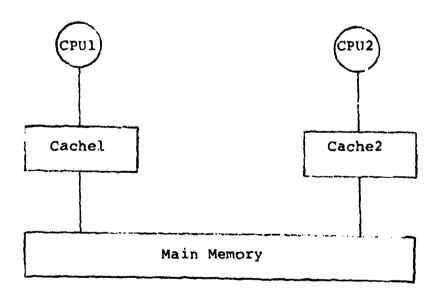
The CPU can only access words that are in the cache. If a word that is needed for processing is not already in the cache, it will first have to be transferred from main memory to the cache. Once the word is in the cache it becomes accessable by the CPU and processing can take place. If the CPU then updates (i.e., mocifies) the word, inconsistency between the copy in the cache and the copy in main memory could develop. This depends on the store algorithm used.

If a store through algorithm (in which the cache and main memory are updated simultaneously) is used, inconsistency will not arise. The price paid for that is a decrease in processing speed as store operations become limited by the speed of main memory. When this price is too high, store-behind or store-replacement algorithms may be used. In both cases main memory is not updated immediately, and as a result inconsistency arises. The modified word in the cache will, for "some" interval of time, be different from its unmodified version in main memory.

The more interesting case, however, is that of multicache systems. Consider the two-cache organization of Figure 1(b). What is important to emphasize here, is that the <u>store-through</u> algorithm is no longer sufficient to avoid inconsistency. Assume, for example, a word whose main memory address is (A), and whose current value is (V) is present in both cachel and cache2. CPUl then



(a) Single-Cache Organization



(b) Two-Cache Organization

Figure (1)

modifies the value of the word in its cachel to (V.), and assuming a store-through algorithm is used, main memory is simultaneously updated. However, cachel continues to have the unmodified version (V). If, before this inconsistency is resolved by either updating, replacing, or invalidating cachels copy, CPU2 attempts to access the word (A), it will get what is now an invalid value (V) from its cachel.

It is time now to adopt what we believe is a more useful definition of what a "consistency problem" is. We claim that inconsistency per se is not necessarily a problem. Reconsider the case of a single-cache organization. We have already explained how inconsistency can arise for "some" interval of time when the store through algorithm is not used. During that interval of time the cache will contain the modified version of the word, while main memory will not. If, during this interval, the CPU needs to re-access the word for processing, what will happen? It will check the cache, find the word in it, and, therefore, access it i.e., no transfer from main memory will be needed. Thus, although inconsistency exists, no problem arises because the CPU will always access the updated version of the word from the cache.

With this in mind, we now adopt the following definition of a "consistency problem" (Censier and Feautrier, 1978):

"A consistency problem exists in a cache-based system if the value accessed by a CPU is not the value given by the latest store operation (by any CPU) to the same address."

It is obvious, from the above discussion, that there will be no consistency problem for single-cache organizations. These, unfortunately, are not very attractive for high performance systems.

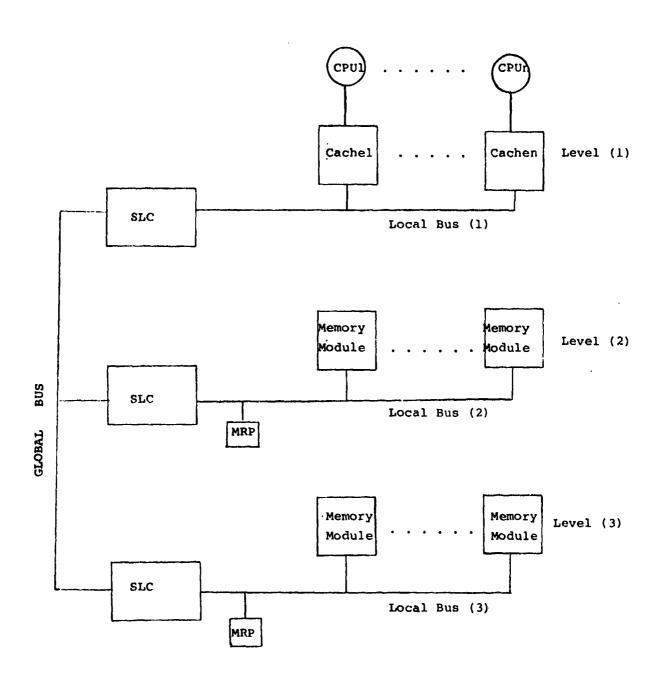
In the next part of this paper we present a brief discussion of INFOPLEX, a highly parallel multi-processor computer system that utilizes a multicache organization. In such an organization the consistency problem is a significant one. The INFOPLEX organization will constitute the context within which we shall evaluate the different approaches for handling the multicache-consistency problem.

#### II. INFOPLEX: A MULTI-PROCESSOR COMPUTER SYSTEM

A research project is currently underway at the MIT Sloan School of Management aimed at investigating the architecture of a new data base computer, called INFOPLEX, which is particularly suitable for large-scale information management (Madnick, 1979). The specific objectives of the project include providing substantial performance improvements over conventional architectures, supporting very large complex data bases, and providing extremely high reliability.

To provide a high performance, highly reliable, and large capacity storage system, INFOPLEX makes use of an automatically managed memory hierarchy. It is this aspect of the project that will be of relevance in our present discussion.

As a simplistic illustration, we show in Figure 2 three levels (only) of the memory hierarchy. As can be seen, this is a multicache organization. The proposed number of caches (m) is relatively large compared to present day systems (e.g., m = 32). For the INFOPLEX



SLC : Storage Level Controller
MRP : Memory Request Processor

Figure (2)

objectives such a large number of caches is essential. It will, for example, help attain the high performance improvements sought (up to a 1000 fold increase in throughput over conventional architectures). In addition, it allows for the implementation of such features as dynamic reconfiguration and automatic recovery which are aimed at improving the reliability of the system.

Two important "boxes" in the design of Figure 2 are the Storage Level Controller (SLC) and the Memory Request Processor (MRP). The function of the SLC is to couple the local bus of a storage level to the global bus that connects all storage levels. In essence, the SLC serves as a gateway between levels. For example, the SLC of level (1) accepts requests to the lower storage levels from the caches and forwards them to the SLC of level (2). When the responses to these requests are ready, the level (1) SLC accepts them and sends them back to the appropriate caches.

The Memory Request Processor (MRP) performs such functions as: implementing the storage management algorithms (e.g., directing the transfer of information across a storage level); handling all the communication protocols that are peculiar to the particular storage modules (devices) at a storage level; and mapping virtual addresses into their real equivalents. (Note that an MRP is not needed at the cache level.)

The INFOPLEX organization described (briefly) above will constitute the context within which we shall study the different approaches for handling the multicache-consistency problem. For

further information on INFOPLEX the interested reader can consult the following references: (Machick, 1975; Machick, 1979; Lam, 1979; Lam, and Machick, 1979; and Hsu, 1980).

# III. THREE COMMON APPROACHES FOR SOLVING THE CACHE-CONSISTENCY PROBLEM

In Part (I) we explained how a consistency problem could arise in multicache memory systems. We saw, for example, that in the two-cache organization of Figure 1(b) a word (A) that existed in both cachel and cache2 could be modified to (V.) in cachel and in main memory but not in cache2, and thus giving rise to an inconsistent state. This example, although rather simple, is quite adequate to demonstrate the motivations behind the basic strategies that have been used to handle the consistency problem in multicache systems. There are two such strategies. First, we could restrict the "encacheability" of data items, such that only those data items that cannot cause inconsistencies are allowed to move into the cache level. For example, words that can only be READ would be encacheable. On the other hand, all data items that could potentially cause inconsistencies are prohibited from moving into the cache level, and thus all accesses to them are done through main memory. Word (A) of the above example would, therefore, fall in this category, and so it (under this

strategy) would have been prohibited from moving into either cachel or cache2. Thus accesses to word (A) by both CPUl and CPU2 would have been made to its <u>single</u> copy in main memory, and no inconsistency would have resulted. The price paid, however, is that accesses to word (A) are now done at main memory speed and not at the faster cache speed.

The second basic strategy that has been employed doesn.t put any such restrictions on moving data items into the cache level. The idea here is to "invalidate" a cache line when there is a risk that its contents have been modified elsewhere in the system. When a cache line is invalidated (by setting, for example, a flag in the cache directory) it is considered not in the cache. Referring again to the above example, when the value of word (A) is modified in cachel (and in main memory) to (V.), word (A) in cache2 is invalidated. Thus if CPU2 happens to request word (A) at a later time, it will have to access it from main memory since the invalidated version (V) in its cache is considered not to exist. CPU2 will, therefore, access the valid value (V.).

In the remainder of this section we will present more specific approaches to handle the consistency problem in cache-based systems. In particular, three approaches that are proposed in the literature will be discussed, namely, the "Broadcasting," the "Store-Controller," and the "Multics" approaches. The "Broadcasting" and "Store-Controller" approaches are based on the second strategy discussed above. They, though, implement it differently. The "Multics" approach, on the other hand, is based on the first strategy.

## III.1. The "Broadcasting" Approach

The idea here (as mentioned in the second strategy above) is to invalidate a cache line when its contents is modified in another cache in the system. When a cache line is modified its address is broadcasted throughout the system so that other caches sharing the line would invalidate their now outdated version of it.

Every cache is connected to an auxiliary data path over which all other caches send the addresses of lines to be modified. Each cache constantly monitors this path and executes a searching algorithm on all addresses thus received. In case of a "hit," the affected line is invalidated.

When a CPU needs to read (or write) a word that doesn.t exist in its own cache, the word will be seized from main memory. To ensure consistency main memory must always be kept "up-to-date." This (in general) can be guaranteed only if a store-through algorithm (in which the cache and main memory are updated simultaneously) is used. As was argued before, such a restriction is not without its cost: A decrease in processing speed as store operations become limited by the speed of main memory.

Another major drawback of this approach is that the invalidation data path must accompodate a very high traffic. The mean write rate for most processor architectures lies in the range between 10 and 30 per cent (Censier and Feautrier, 1978), and thus if the number of processors is higher than two, the productive traffic between a cache

and its associated processor may be lower than the parasitic traffic between the cache and all other caches. This explains why this approach has been confined to systems with at most two caches (Censier and Feautrier, 1978).

#### III.2. The "Store-Controller" Approach:

The basic idea here is the same as in the "Broadcasting" approach i.e. to invalidate a cache line when its contents have been modified elsewhere in the system. It is in the implementation that the two approaches differ. Here, a "Store-Controller" SC (see Figure 3(a)) is used at the cache level to keep track of every line in every cache. The store-controller "knows," not only which lines are in which cache, but also which caches share any single line. When, therefore, a line that is shared by two or more caches is updated (i.e., modified) in one of them we do not now need to broadcast invalidation requests to all caches. We, instead, use the information in the store-controller to send invalidation requests to only those caches that are sharing the updated line, if any. In other words, the motivation behind using the store-controller is to filter out all unnecessary invalidation requests.

We will present, in a flowcharted form, an example implementation of this approach which is largely based on Tang.s proposals (Tang, 1976). In this implementation, if a processor wants to write and the line is not found in its cache, then the line is always brought to the

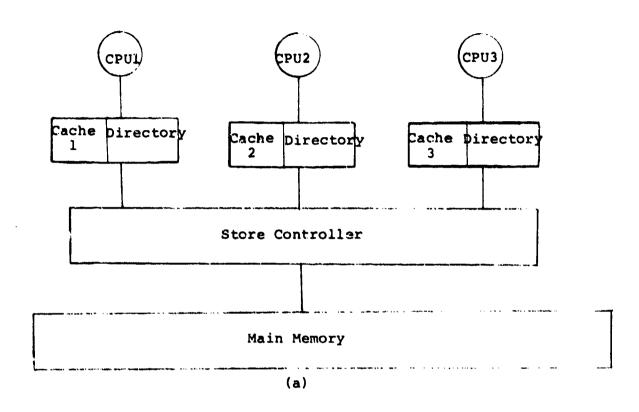
algorithm used is the "store-replacement" algorithm. This means that when a line is modified in a cache, main memory is not concurrently updated. It is updated later when the line has to be replaced in the cache or when other caches need to have the line.

Figure 3(a) shows how the store-controller fits conceptually into the cache organization. Figures 3(b) and 3(c) show possible layouts for the directories of both the cache and the store-controller. The "status" column of the cache directory shown in Figure 3(b) needs some explanation. The status of a line can be one of three things:

- 1. Private: For a line which has been modified (with respect to main memory) or is going to be modified. A private line exists in only one cache.
- 2. Non-Private: For a line that exists in one or more caches and which has not been modified with respect to main memory.
- 3. <u>Invalid</u>: For a line that has been modified elsewhere and thus becomes outdated. An invalid line is considered "not in the cache."

In Figures 4 and 5 the READ and WRITE operations are illustrated respectively.

The two major drawbacks of implementing this approach in a highly parallel multi-processor computer system such as INFOPLEX where the number of caches is relatively large are:

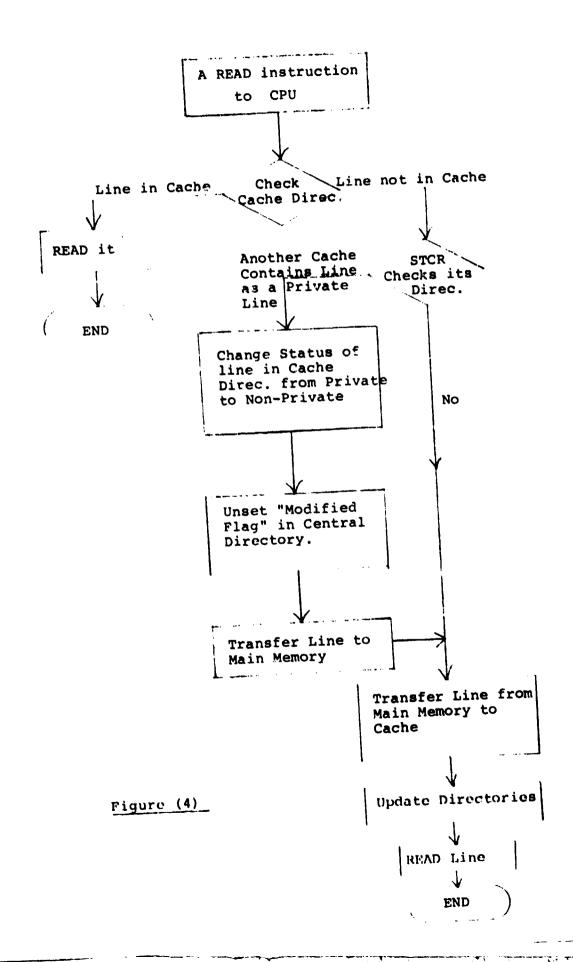


Main	Memory	Addres	Cache Line Location	Status
·				
		Ì		i i
L.		(h.)	Cache Directory	

Main Mem. Addr.	Cachel Cache2	Modified Flag
	One bit per cache is set as follows:	Set if line is a pri-
	l if line in cac 0 if not	e vate lite in a cache

(c) Central Directory

Figure (3)



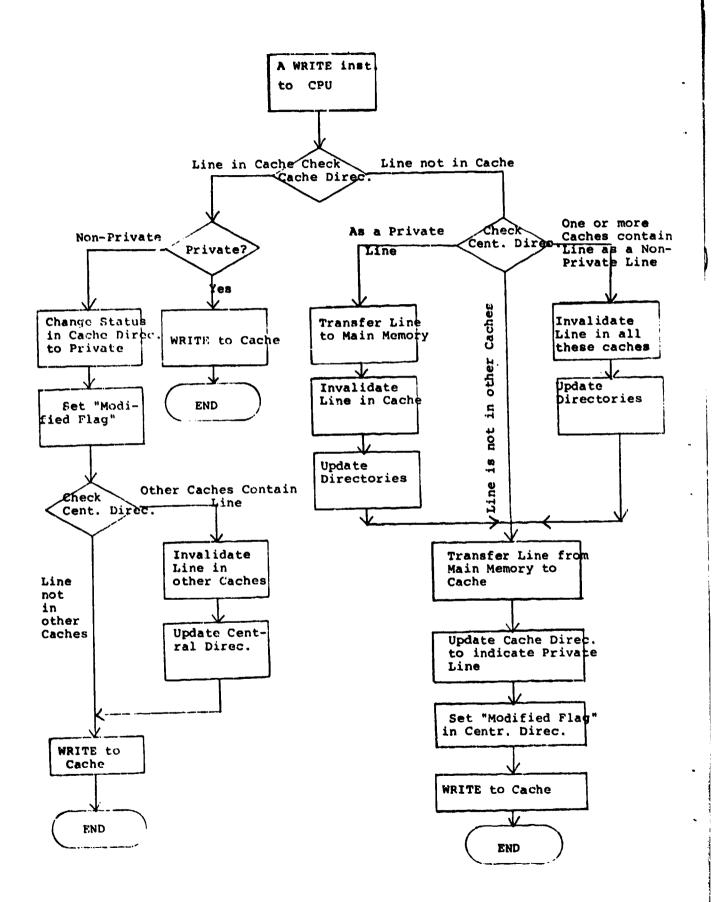


Figure 5

- 1. The size of the central directory becomes too large, which means increased time in processing it and increased costs in building it.
- 2. The store-controller could become a bottleneck in the system as the traffic between itself and the caches becomes very large.

## III.3. The "MULTICS" Approach

This approach, which is used in Honeywell.s Multics computer system (Greenberg, 1978), has two important features. Firstly, the Multics cache is a "store through" cache. This means that the cache and main memory are updated simultaneously. The second feature is that the system address space is divided into <u>segments</u>, each of which has associated with it names and per-user access rights which govern the ability of each potential user of the segment to read and/or write its contents (i.e., words).

Every segment is known by the system to be either "writable" or "non-writable." The non-writable class of segments, that is those to which no users have write access, is an important and statistically significant one in MULTICS. All procedures, including all parts of the operating system, utilities, libraries, translators, and so forth, fall into this category. These segments are "encacheable" by all processors. This means that their contents (or words) are allowed to "migrate" to any or all caches. Notice that when such words find their way into a cache (or more than one cache) there is no possibility for a

consistency problem to arise (for such words) since no processor can write or modify them.

For the other class of segments, those which are writable, the situation is slightly more complicated. For a segment falling in this category, there are three possible states:

- 1. One or more processes (users) are accessing the segment but none of them has a write access to it. The segment is encacheable by all processors but no consistency problem will arise.
- 2. One or more processes are accessing the segment, with at least one of them having write access. The segment becomes non-encacheable i.e., its words cannot migrate to any of the caches. The consistency problem will not arise here also since there will only be one copy (i.e., the one in main memory).
- 3. Only one process that has write access is accessing the segment. The segment is encacheable only to that process. And it is only in this third state that the consistency problem could possibly arise. Consider the following scenario:

Assume a certain segment Sl is addressable by only one process PROC1 which is currently running for the first time on processor CPU1. Assume also that PROC1 has write access to Sl. Thus Sl is encacheable by CPU1. As long as CPU1 runs PROC1, words of Sl may be drawn into CPU1.s cache and be modified by CPU1 with no problem. During this period, no other processor can address Sl, for by assumption it is addressable only by PROC1, which is uniquely associated with

CPUI during the interval in question. Thus, it is impossible for other processors to draw words of Sl into their caches as long as PROCl is associated with CPUl. Until CPUl leaves PROC1, there is thus no danger of words of S1 in CPU1.s cache becoming outdated, as no other processor can address S1. Similarly, there cannot be words of Sl in any other processor.s cache, for by assumption, CPUl was the first and only processor to run PROC1. Thus, there is no danger that modifications to words of S1 made by CPUl can invalidate copies in other processors. caches, since such copies cannot exist. Potential difficulty arises when CPU1 has left PROC1, and some other processor attempts to run PROC1. The first time this happens, there is no problem. Since all words in main memory are accurate, by virtue of the store-through cache, another processor, say CPU2, cannot have inaccurate data, or main memory is accurate, and we have just shown how CPU2.s cache may not contain inaccurate data. However, while PROC1 runs on CPU2, CPU2 may modify words of S1 in its own cache and in main memory. Still there is no problem. Main memory is accurate, as is CPU2.s cache. This can go on like this as long as processors which have never ran PRCC1 (since PROC1 started running) run it. However, the first time some processor which has already ran PROC1 since then attempts to run it again, the scheme appears to break down. Assume CPUL attempts to run PROC1 for the second time. There may be words of S1 in CPUl.s cache from the previous time CPUl ran PROC1. Some of these words may have been modified by PROC1 while it ran on CPU2. Thus, these words are accurate in main memory and in CPU2.s cache, but are inaccurate in CPU1.s cache, for CPU2 had no way of knowing or acting upon the fact that they were in CPU1.s cache.

The MULTICS solution to "its" consistency problem is simple: clear the cache of a processor upon entering a process if it was not the last processor to run that process. This is performed by the MULTICS process dispatcher, with a special processor instruction that accomplishes this task. This ensures that no words of any per-process writable segment will be found in a processor.s cache if there was any possibility that any of those segments may have been modified by other processors. The operating system maintains in the control block describing each process the identity of the last processor to have run this process thus this check is easy to make when a processor is dispatched into a process.

From an INFOPLEX-type-system view point, there are three major drawbacks to the MULTICS approach, all of which are performance related:

1. The class of segments that are non-encacheable can be of significant size, and thus dampening the performance gains sought by the cache organization. Note that in MULTICS there is the significant class of segments which we termed "non-writable" and which are encacheable. In data base computers, like INFOPLEX, this category which contains things like utilities, libraries, and translators will probably be much smaller, and thus decreasing the portion of encacheable segments. (There could, however, be

special applications where this doesn.t apply e.g., READ-only data base applications.) In addition, the MULTICS approach doesn.t discriminate between two processes who although both have write access to a segment, one actually exercises the "right" and modifies the segment, while the other doesn.t. In both cases the segment will become non-encacheable if there are other processes that are also reading it. This means that in both cases the system.s speed will be slowed down to the speed of main memory.

- 2. Using a store-through algorithm has its own performance disadvantages. As was stated earlier, it limits the speed of store operations to that of main memory, and thus defeating the very purpose of using a cache.
- 3. The procedure of clearing up the cache is also a wasteful one. Note that a cache is always cleared if its processor wasn.t the last one to run the process. All access requests to the cleared cache contents that would have otherwise been serviced by the cache, must now wait for transfers from main meemory. This will, obviously, slow down the system.

#### III.4. Conclusion

We have analyzed the three common approaches that have been proposed in the literature to handle the consistency problem in multicache systems. We have considered each approach in the context of the INFOPLEX framework presented earlier. Within this context we were able to identify some major drawbacks in each of the approaches.

In the next section we propose a new approach to handle the cache-consistency problem in multi-processor architectures. In Part (V) we will evaluate the performance of this proposed approach.

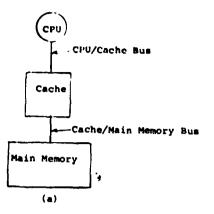
IV. THE "COMMON-CACHE / PENDED TRANSACTION BUS" APPROACH

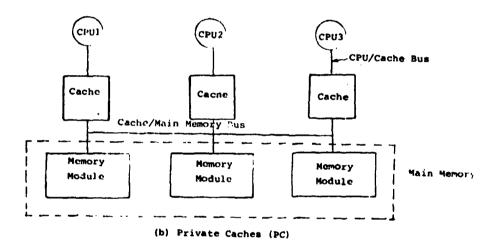
#### IV.1 Introduction

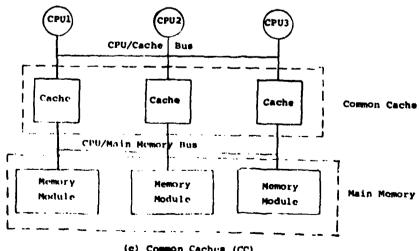
We argued in the beginning of Part (I) that in a single-cache memory system (Figure 6(a)), where there is only one access path between each level, no cache-consistency problem will arise. Once two or more caches are used, however, the potential for the problem develops.

The "traditional" approach in employing caches in multi-processor systems has been to basically replicate the structure of Figure 6(a) for each of the processors, as shown in Figure 6(b), and then solve any problems that arise. A problem that arises, of course, is the cache-consistency problem, and the three basic approaches that have been developed to handle it are those of Part (III).

Implementing any of the three approaches will obviously constitute some processing overhead. As an example, consider the







(c) Common Caches (CC)

Figure (6)

"Store-Controller" approach and refer in particular to the flow-chart of Figure 5. When a CPU needs to modify a line that exists in its cache, and the line happens to be a "non-private" line, then the status of the line is first changed to "private" and the "modified flag" is set. Next, the Store-Controller.s directory is checked, and if the line is found not to be shared by other caches it is modified. If, however, it happens to be shared, then messages are sent to the appropriate caches to invalidate the line, the central directory is updated, and finally the line is modified. How much overhead did we incur to maintain consistency? Well, compare the above steps with those needed for a uniprocessor system where the cache-consistency problem does not arise. In such a system, when the CPU needs to modify a line that exists in its cache, it simply proceeds and modifies it.

None of the above checks, updates and messages are needed.

The concern over the processing overhead needed to maintain consistency is a legitimate one. Such overhead does undoubtedly dampen the performance gains (e.g., system throughput) which are sought by introducing caches in the first place. Attempting to minimize this overhead, therefore, seems an attractive direction for research work.

In this research endeavor, we are basically proposing an architecture that eliminates the processing overhead associated with handling the multicache-consistency problem. The architecture is depicted in Figure 6(c). The important distinction between this organization and that of Figure 6(b) is that, here, each of the cache modules is accessed by, and thus services, all of the processors. Thus, just as main memory is common to and shared by all processors,

the cache level in our proposed architecture is also <u>common</u> to and shared by all processors. In such a scheme there is no need to store more than a single copy of any data item at the cache level. This eliminates redundancy. And with no redundancy at the cache level, no inconsistencies can obviously arise, which in turn eliminates the need for mechanisms to maintain cache-consistency and the overhead associated with such mechanisms.

It is important to note that this architecture preserves the basic intent behind cache-based systems, namely, using a high speed memory level between the CPU and main memory in order to bridge the speed gap between the two. It, however, introduces the concern as to whether the CPU/cache bus (see Figure 6(c)) can handle the needed high volume of traffic. Comparing the Private Cache (PC) architecture of Figure 6(b) against the Common Cache (CC) architecture of 6(c), we note that the CPU/cache bus in PC handles the transaction load generated by a single processor where as in CC it handles the load generated by all the CPUs. We can, therefore, expect that the load on the CPU/cache bus in our proposed (CC) architecture to be close to N-times that of the PC architecture, where N is the number of processors. Of course, the load will be somewhat less than exactly N-times because in PC some overhead traffic will be generated by the mechanism used to handle the cache-consistency problem, and which will not be needed in (CC).

Thus, to re-state, the Common Cache approach eliminates the cache-consistency problem (i.e., by eliminating private caches) but there is a fundamental concern that the CPU/cache bus will develop into a bottleneck that degrades the system.s performance. In the next

section we introduce the Pended Transaction Bus Protocol, which we believe provides the basis for a viable solution to the problem.

#### IV.2 The Pended Transaction Bus (PTB)

The degree of bus utilization of a processor is a function of the physical characteristics of the bus, and the protocol used on it. The physical characteristics of the bus which include the length, voltage levels, impedence, termination, capacitance, noise immunity, and overall reflection characteristics, affect its operating speed.

Ultimately, any bus is limited by the speed of electricity along a conductor (0.6 to 0.9 nanoseconds per foot typically, depending on wire characteristics). Careful electrical analysis and physical layout can optimize these parameters to achieve reasonable electrical speed.

Given an electrical bandwidth of the bus, as formed by the bus wires and the driving logic, the actual data bandwidth becomes a function of the protocol used on it. The traditionally high bus utilizations of multi-microprocessor architectures that employ the single bus as their interconnection scheme is primarily a result of the bus protocol used, and which is called the "master-slave" protocol. In such a protocol, the CPU (master) asserts a request on the bus, and the memory (slave) that receives it does the appropriate action (e.g., a READ), and then responds (with the requested data). The bus is viewed as "busy" during this entire time. A large portion of this time is actually spent waiting for the slave to complete the requested action.

The electrical and logical time to transmit the actual request and return the reply are a relatively small portion of the total bus usage time. The period of time between the request and the acknowledge is a wait interval, and no useful work is done with the bus during this time. Bus utilization could be significantly reduced if we released the bus during the wait interval. To do this we split the transaction into two parts, a request part and a reply part. The master requests the bus, and upon being granted it, sends the request to the slave at maximum speed. The slave acknowledgles reception of the request, and starts to work on it. The processor then releases the bus and waits for the results. When the slave completes its task, it asks for the bus, and upon receiving it sends the reply back to the originating master at maximum speed. The time between the request and reply can be used by other masters and slaves to transfer other messages over the bus. Because the slave stores the incompleted request from its master, it is called a pended transaction, and the bus protocol, developed at M.I.T., is called a Pended Transaction Bus (PTB) protocol (Toong, et al, 1980).

In the above discussions, it was presumed that the slaves were always able to accept the master requests when presented. This would imply that each master was using different slaves to guarantee such separation. Given the shared nature of the cache data, it is likely that two (or more) masters may make requests to the same cache-memory slave. The simplest scheme to resolve this contention problem is for a busy slave to refuse a new request and make the requesting master retry at a later time when the slave becomes free. This, however, is wasteful of bus bandwidth, since the time spent to send a request out

the first time, only to be refused, is not useful. Furthermore, the slave may still be busy when the master tries again later.

Goodrich (Goodrich II, 1980) has proposed putting queues on the inputs of the slaves to buffer requests. That is, any requests that come while the slave is busy would be placed in a first-in-first-out queue, and these requests would be serviced in order when the slave ia able to handle them. Such a scheme would reduce the bus load to what is actually needed for transmission, without any extra cycles. In addition, it would also improve the slave response time as seen by the master over the simple scheme described before, since the slave would have the request in its queue, and would service it as fast as it could, not just when the master is finally successful in transmitting it. Note that a queue overflow need not be fatal in this system. It can be treated like a refusal in the previous scheme, and would simply require the master to retransmit the request later. If the queue size is sufficiently large, such refusals would be rare. Finally, for best slave throughput, there should also be a queue on the output of each slave.

# IV.3 An Application: The INFOPLEX Storage Hierarchy

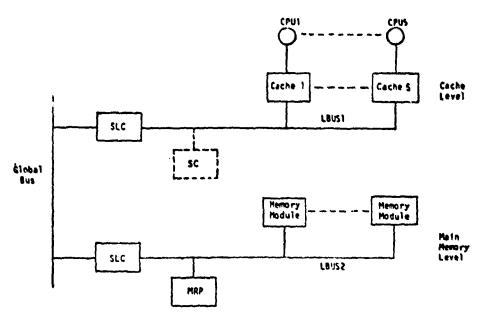
In the above sections we have introduced an approach to the cache-consistency problem in highly parallel multi-processor computer systems, which we will call the "Common-Cache / Pended Transaction Bus" approach (CC/PFB). Its two distinctive features are: Firstly, the

conventional one to one relationship between processors and caches i.e., where each CPU has its own private cache, is replaced by an N-to-M relationship, where a pool of cache-modules is commonly shared by all processors. Secondly, we propose the use of the <u>Pended</u>

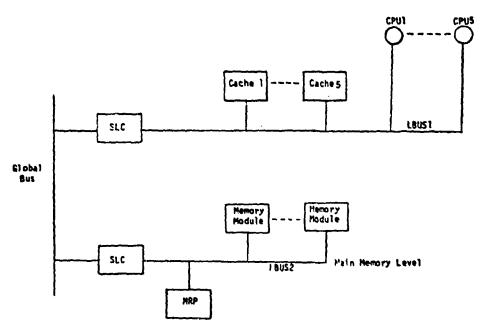
<u>Transaction Bus</u> (PTB) as the interconnection scheme that connects the processors and the cache-modules.

In this section we describe how the CC/PTB architecture can be incorporated into the storage hierarchy of the INFOPLEX data base computer. Schematically the proposed architecture would look like Figure 7(b). We will, honceforth, refer to this architecture as INFOPLEX/CC (for Common Cache) while referring to the original Private Cache organization (shown in Figure 7(a)) as the INFOPLEX/PC architecture.

The key INFOPLEX storage hierarchy operations are the READ and WRITE. In INFOPLEX/PC two strategies are reeded to implement these operations, a strategy for the cache level and another for all other levels. The reason for this is manifested in Figure 7(a), where it can be seen that the organization of the cache level is different from the other levels of the hierarchy. In INFOPLEX/CC, on the other hand, the cache level organization is very similar to that of the lower storage levels. As a result, the strategy used to implement the READ and WRITE operations could be the same for all levels of the hierarchy with very minor provisions to account for the few differences that do still distinguish the cache level. (For example, the absense of an MRP at the cache level.)



#### (a) Private Cache (PC) Architecture



(b) Common Cache (CC) Architecture

LBUS1: Local BUS at level (1) - cache level

LBUS2: Local BUS at level (2) - main memory level

SC: Storage-Controller - needed only if "Storage-Controller" approach is implemented

SLC: Storage Level Controller - it couples a storage level to the Global bus (i.e. serves as a gateway between levels)

MRP: Memory Request Processor - performs address mapping function

Figure (7)

We will attempt now to explain briefly how the basic READ and WRITE operations will be implemented in the INFOPLEX/CC architecture. To a large extent this will be based on the work of Lam (Lam, 79), where a much more detailed and complete discussion is presented.

All READ and WRITE operations are performed in the highest storage level L(l) i.e., the cache level. If a referenced data item is not in L(l), it is brought up to L(l) from a lower storage level via a READ-THROUGH operation. The effect of an update to a data item in L(l) is later propagated down to the lower storage levels via a number of STORE-BEHIND operations.

When a READ request is issued by a processor, the cache level is checked to see if the requested data is in it. If the data is found in a cache-module, it is retrieved and returned to the processor. If, however, the requested data is not found in the cache level, a READ-THROUGH request is queued to be sent to the next lower level L(2) via the Storage Level Controller (SLC). As mentioned in Part (II) the SLC serves as a gateway between the storage levels of the hierarchy.

At a storage level, a READ-THROUGH request is handled by the Memory Request Processor (MRP). An MRP performs the address mapping function. It contains a directory of all the data maintained in the storage level. Using this directory, the MRP can determine if the requested data is in one of the storage devices at that level. If the data is not in the storage level, the READ-THROUGH request is queued to be sent to the next lower storage level via the Storage Level Controller (SLC).

If the data is found in a storage level L(i), the MP maps the main memory address of the requested data item into its real address in L(i). This real address is used by the appropriate storage device to retrieve the block containing the data and then passes it to the SLC. The SLC would then broadcast the block to all upper storage levels by dividing it into fixed size packets. Each upper storage level has a buffer to receive these packets. A storage level only collects those packets that assemble into a sub-block of an appropriate size (peculiar to the storage level) that contains the requested data. This sub-block is then stored in a storage device. At L(l), the sub-block (i.e., the cache line in this case) containing the requested data is stored, and the data is finally sent to the processor that initiated the request.

In a <u>WRITE</u> operation, the data item is written into a cache-module, and the processor is notified of the completion of the WRITE operation. We shall assume that the data item to be written is already in L(1). (This can be realized by reading the data item into L(1) before the WRITE operation.) A STORE-BEHIND operation is next generated by the cache-module and sent to the next lower storage level. INFOPLEX uses a two-level STORE-BEHIND strategy. This strategy ensures that in a hierarchy with N levels, an updated block will not be considered for eviction from a storage level L(i), until its "parent" blocks at levels L(i+1) and L(i+2) are updated. This scheme will ensure that at least two copies of the updated data exist in the storage hierarchy at any time. The motivation behind using such a strategy is two-fold. Firstly, the reliability of the system is enhanced because at least two copies of newly written data are always maintained until the data is "securely" copied at the lowest level of

the hierarchy. Furthermore, the STORE-BEHIND strategy allows for the updating of lower storage levels to be carried out at slack periods of system operation, thus enhancing performance.

In the above discussions we didn.t show how a specific cache-module can be correctly selected to handle a READ or a WRITE operation of a particular data item. In all but the cache level this function is performed by the Memory Request Processor (MRP), which maps main memory addresses into their physical equivalents at a particular level. What we need, therefore, is to augment the processors/common-cache interface to translate main memory addresses to physical addresses in the cache-modules.

There are two possible places to perform the translation operation: at the processor interface and at the cache interface. At the processor interface, the address gets translated before it reaches the bus. This requires that each processor be "informed" about all current lines at the cache level. This information would be used to translate all the main memory addresses of these lines to their equivalents at the cache level. An advantage of this is that the translation can be performed while the procesor is arbitrating for the bus, and if the translation operation is fast enough, it can be done without any access time penalty.

In the second possible scheme the address gets translated at the cache-module interface. What would be needed here is a mechanism incorporated in the recognition circuitry of each cache-module that would translate the main memory address put on the bus, and that would

accordingly decide IF the desired data item is present in the cache level and if so WHERE i.e. in which cache-module and in which location in it. Both these should be done as fast as possible. Tag directory schemes incorporating set associative mapping are suggested by Mattick (Mattick, 1977).

In Part (V) we will evaluate the performance of both these schemes when implemented at the cache level of the INFOPLEX storage hierarchy.

# V. EVALUATING THE PERFORMANCE OF THE "COMMON-CACHE / PENDED TRANSACTION BUS" APPROACH

## V.1. Introduction

To evaluate the performance of our CC/PTB approach we used the INFOPLEX multi-level storage system as our test case. Very slight modifications to the existing design were needed to incorporate "CC/PTB" into the INFOPLEX storage stucture. For example, instead of using the four level memory hierarchy studied previously by Lam (Lam, 1979a) just two levels, the cache level and main memory were sufficient for our purposes.

The "CC/PTB" approach was compared against two benchmarks.

Firstly, we evaluated the performance of the "Store-Controller" approach as a representative of the traditional approaches. Selecting the "Store-Controller" approach to evaluate our scheme against cannot, however, provide an effective answer to the question of how "optimal" either approach is. What is needed is an "absolute" benchmark against

which both approaches could be judged. For this purpose we evaluated the performance of a traditional private cache architecture assuming no overhead for handling the cache-consistency problem. This, then, constitutes the "performance ceiling" that no cache-consistency handling mechanism (for INFOPLEX) could possibly exceed. It is also a valuable reference point in that it tells us how close we are to an "optimal" solution.

# V.2. The Evaluation Tool

The evaluation was performed by producing a simulation model in GPSS. We developed four separate GPSS programs:

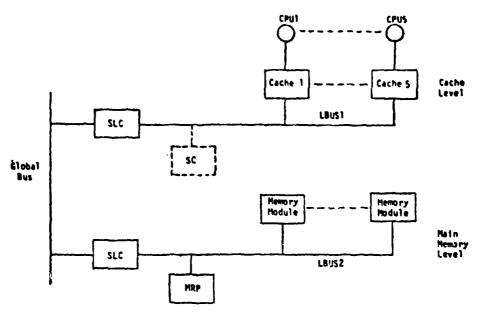
- 1. Program "OPT" ignores the cache-consistency problem, and thus incorporates no overhead for handling it. It provides us with a ceiling on performance.
- 2. Program "STCR" incorporates the "Store-Controller" approach.
- 3. Program "CC/PTB/C" incorporates the "CC/PTB" approach with the translation operation performed at the cache-interface.
- and 4. Program "CC/PTB/P" incorporates the "CC/PTB" approach with the translation operation performed at the processor interface.

The GPSS code for each of the above four programs is presented in a separate Appendix (Appendices I through IV). All four programs are highly detailed simulators. A widely used index that characterizes the degree of detail of a simulation model is its resolution in time, which

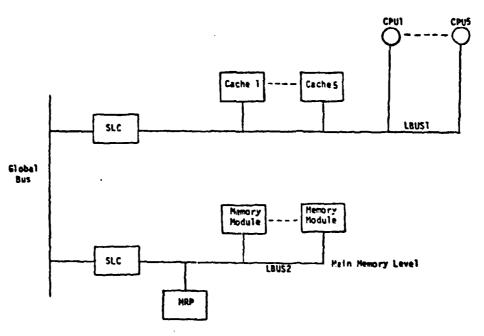
is defined as the shortest interval of simulated time between two consecutive events being considered (Ferrari, 1978). In our four simulations, the resolution in time is 10 nanoseconds. Such detailed simulators are likely to be more accurate and to have broader field of application than less detailed ones. However, they are certainly more expensive to design, implement, test, document, and use.

In addition to deciding on the degree of detail of the models, another basic decision had to be made concerning the workload that would drive the simulations. We decided to perform our measurements in an operating environment that is not at all uncommon in present-day computer systems, namely, running at capacity. Under such conditions, there will always be at least one transaction in the system.s input queue(s) waiting to be serviced. In such a case, the performance characteristics that are of interest to us, such as throughput per unit time, become insensitive to the distribution of job arrivals.

Before concluding this section we would like to emphasize some of the structural differences between the four models, as well as some of their common characteristics. The basic structural differences are highlighted in the diagrams of Figure 8. In Figure 8(a) a two-storage-level version of the INFOPLEX storage hierarchy proposed in (Lam, 1979) is shown. To support the "Store-Controller" approach an SC "box" (dotted in Figure 8(a)) is added to the architecture. In Figure 8(b) the architecture we proposed in Part IV to support the two versions of the "CC/PTB" approach is incorporated into the INFOPLEX storage system. Notice that in the CC/PTB architecture we deliberately maintained the same number of caches (5) as in Figure 8(a). It is



## (a) Private Cache (PC) Architecture



(b) Common Cache (CC) Architecture

LBUS1: Local BUS at level (1) - cache level

LBUS2: Local BUS at level (2) - main memory level

SC: Storage-Controller - needed only if "Storage-Controller" approach
is implemented

SLC: Storage Level Controller - it couples a storage level to the
Global bus (1.e. serves as a gateway between levels)

MRP: Memory Request Processor - performs address mapping function

Figure (8)

important to realize that while a 1:1 relationship between the CPUs and the caches is inherent in the Private Cache (PC) architectures, such is not the case for the Common Cache (CC) architectures. We, however, chose to maintain the same number of caches in both architectures (and four simulation models) to neutralize it as a factor that might affect performance.

Finally, there is a set of common characteristics that are shared by all four models, these are:

Degree of Multiprogramming of a CPU = 10

Bus Width = 8 bytes

Size of Transaction without data = 8 bytes

Size of Transaction with data:

at Level 1 = 8 bytes at Level 2 = 64 bytes

Size of Data Buffers = 10 64-byte transactions

Finally, the Pended Transaction Bus (PTB) protocol will be used in all three architectures (and four models). That is, we are committed to the PTB protocol in INFOPLEX as a result of our experimental work (at M.I.T.), which demostrated its performance advantage.

### V.3. The Simulation Experiment:

which will also serve as our dependent variable, will be the total system throughput as measured by the total number of transactions processed per unit of time. As for the independent variable, there

were two candidates: (1) the hit ratio, which is the percentage of time that a referenced data item is found in the cache level; and (2) the transaction mix i.e., the percentage of READ requests versus WRITE requests.

The latter was chosen because we felt it is in a sense, a more independent variable. What we mean is that the transaction mix is largely a function of the use of the system and as such we have very little control over it. The hit ratio, on the other hand, is system-dependent. It can be affected by manipulating such system parameters as the total size of the cache level and the size of the individual cache blocks.

Thus, the hit ratio will be held constant, throughout the experiment, at a value of 0.90 for READ requests and 1.00 for WRITE requests. (That is, we are assuming that a WRITE of a data item is always preceded by a READ to it.) The transaction mix i.e., the percentage of READs, will be allowed to vary in the range from 70 % to 90 %. This is the range in which the mean READ rate lies for most processor architectures (Censier and Feautrier, 1978).

Both the hit ratio and the transaction mix are variables that affect the performance of the system but which are independent of the mechanisms used to handle the cache-consistency problem. There are, however, variables that are peculiar to the particular mechanism used, and which influence the system.s performance significantly.

In the "Store-Controller" approach the degree of sharing between

the caches is by far the most important such variable. We evaluated the "Store-Controller" approach under two operational modes, a "pessimistic" mode and an "optimistic" mode. Under the optimistic mode no sharing between caches takes place, yielding an upper bound on the performance of this approach. Under the pessimistic mode a high degree of sharing will be introduced (50 % of the cache blocks will be shared by more than one cache-module). This will then provide us with a conservative lower bound on the performance of the "Store-Controller" approach.

With respect to the "CC/PTB" approach we vill also follow the above strategy, and evaluate it under both "pessimistic" and "optimistic" conditions. Under the optimistic condition we will assume the load on the cache-modules to be uniformly distributed (i.e., each of the 5 cache-modules carries 20% of the load). And for the pessimistic case we will assume the load to be linearly distributed between 10% at the least loaded cache-module and 30% at the most loaded.

### V.4. The Hardware Parameters

It is necessary to determine the speeds of the different hardware components in the INFOPLEX storage hierarchy. In particular, what we sought were the best possible 1985 projections for these speeds, since the first hardware prototype of the INFOPLEX data base computer was not expected before then.

The forecasts shown below constitute a realistic, but ambitious, scenario for 1985. In other words, they incorporate the fastest possible components that we envision as being available (and appropriate) for building an INFOPLEX in 1985. (Only a subset of the parameters are shown.) The bus speed (b) is 10 nanoseconds, the cache READ/WRITE speed (c) is 20 nanoseconds, and the remaining parameters are multiples of the latter, as shown. In other words, we used the cache speed as a logical building block to "build" the forecasts of the other hardware components (other than the bus). For example, if the cache READ/WRITE speed is 20 nanoseconds the READ/WRITE speed of main memory would be 10 x c = 200 nanoseconds. The parameter values are:

Bus speed (b)	=	10 1	nanoseconds
Cache READ/WRITE speed (c)	=	20	10
Directory Lookup (2c)	=	40	Ħ
Directory Update (4c)	=	80	H
Storage Level Controller (SLC) speed (2c)	×	40	H
Main Memory READ/WRITE speed (10c)	=	200	19

Three other scenarios will be tested. The bus speed, in all three, will remain at 10 nanoseconds. The cache READ/WRITE speed, however, will take the increasing values of 40, 60, and 80 nanoseconds. And finally, the remaining parameters will maintain their relative values in terms of n, the cache READ/WRITE speed. For example, when the cache READ/WRITE speed (c) becomes 40 nanoseconds the READ/WRITE speed of main memory will be 10 x c = 400 nanoseconds.

Selecting several "good" 1985 scenarios reflects the fact that different options, in building an INFOPLEX, will be available to accompdate the cost/performance tradeoffs. And because the bulk of the

system.s cost lies largely in the storage components, the variations in the forecasts between the different scenarios involved mainly those camponents.

# V.5 Analysis of the Simulation Results:

As mentioned previously, our dependent variable in this experiment is system throughput. It is also the criterion we use to evaluate the performance of the cache-consistency handling mechanisms.

The throughput of any computer system is bounded by one of two factors, namely, bottlenecks in the system or the transaction load on it. In section V.2 we mentioned that our models will operate in a maximum load closed-loop environment, where new transactions are continuously generated to replace serviced ones. In such an environment, bottlenecks will definitely arise, limiting the throughput of the system.

Thus, in the process of interpreting the simulation results, we wish to identity and analyze system bottlenecks. In such an analysis, one needs to consider four major factors that directly influence the evolution of a particular system component into becoming the system.s bottleneck. The four factors are:

1. Architecture: Consider for example the (a) PC and (b) CC architectures of Figure 8. In PC the local bus of level 1 (the

cache level) handles only the communications between the five caches and main memory. In the "CC/PTB" architecture the cache level bus must handle, in addition, the communications between <u>all</u> the processors and the cache-modules. The chances for the local bus of level 1 (LBUS1) to become the system bottleneck are, therefore, much higher in the CC architecture than they are in PC.

- 2. Algorithms and Protocols: The set of algorithms supported by an architecture and the protocols and mechanisms used to implement them undoubtedly influence the utilization patterns of the different architectural components. Consider for example the central role of the Store-Controller "box" in implementing the algorithms that support the READ and WRITE operations in the "Store-Controller" approach. Such a role will inevitably lead to high levels of utilization of the Store-Controller.
- 3. Workload: We mentioned in section V.3 that we intend to evaluate the "CC/PTB" approach using two different load distributions on the cache-modules, a uniform distribution and a linear one. In the latter case, the cache-module carrying the highest load (i.e., 30 % of total load) could clearly develop into a system bottleneck.
- 4. Hardware Components. Characteristics: The characteristic of significance here is speed. For an example we refer again to Figure 8. All communications between level 1 and level 2 in the INFOPLEX storage hierarchy go through the Storage Level Controllers (SICs) of both levels as well as through the Global Bus. The time needed by the Global Bus to process any of the communication messages (i.e., the time it takes to transmit the message) will usually be less than that needed by the SIC to

process the same message. This means that the SIC will always saturate before the Global Bus can develop into a bottleneck.

As part of the standard GPSS output, the utilizations of all hardware components (that are modelled as GPSS "facilities") are printed. This allows us to precisely identify the system bottleneck(s). An example is shown in Figure 9. In this case the bottleneck is LBUS1 (the local bus at level (1)) with a utilization of approximately 100 %.

In Figure 10 our simulation results for the four scenarios are presented. In the discussion that follows, we will identify the different scenarios by their cache speed/bus speed ratios (n) (i.e., n = 2, 4, 6, or 8) as it is a convenient parameter that completely characterizes each.

For each technology scenario (i.e., n value) seven curves are plotted. The single solid (——) curve portrays the performance of the "OPT" model, in which no mechanisms for handling the cache-consistency problem (and, therefore, no overheads) are incorporated. Thus the throughput of the "OPT" model, as is demonstrated in the figure, provides a ceiling for all the other models. The "Store-Controller" model.s results are depicted by the two dash-and-dot (——) curves (S1) and (S2). Curve (S1), which is always the dominating curve, is for the case where there is no data sharing between the caches, and (S2) is when 50 % data sharing is introduced. And finally, there are two curves for each of the two implementations of the "CC/PTB" approach. The two dashed (——) curves (P1) and (P2) belong to the "CC/PTB/P"

FACILITY	A VERAGE UTI LIZATION	NUMBER Entries	AVERAGE	SEIZING	PREEMITING
GB78	.474	1199	TIME/TRAN	TRANS. NO.	TRANS. NO.
LBUST	> .999	6514	3.959 1.535	59	
LBUS2	.702	1702	4.126	72	
DRP11	.396	679	5.846		
DRF12	.405	695	5.833	37	
DR 0 1 3	.411	703	5.849		
DRP14	.369	629	5.879		
UR + 15	.398	681	5.856		
KRP1	.479	1 199			
KRP2	. 485	1214	4.000 4.000		
RSP2	.297	719	3.997	<b></b> .	
DRP21	.198	496	4.000	52	

Figure (9)

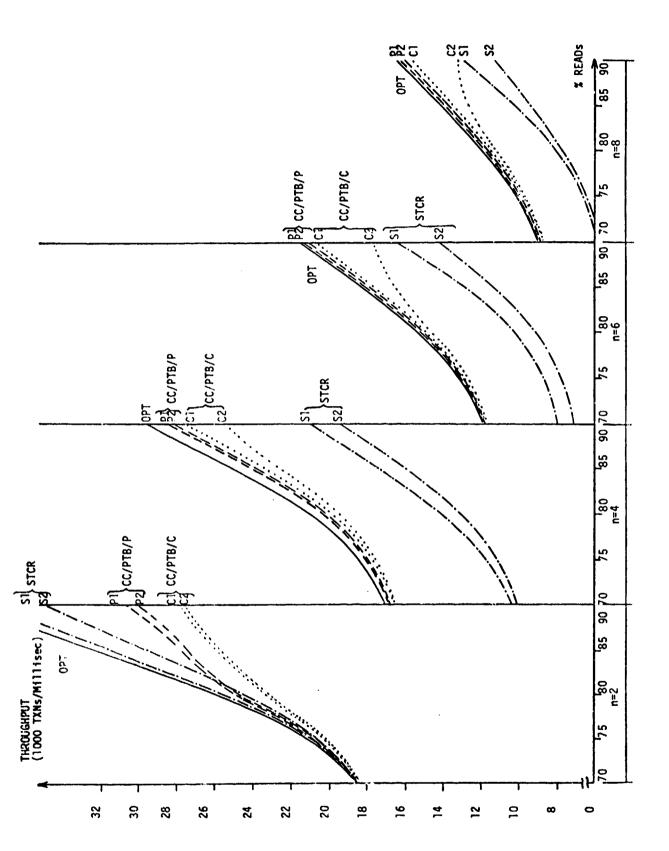


Figure (10)

implementation, while the two dotted (...) ones (C1) and (C2) are for "CC/PTB/C." The two dominating curves in both implementations, namely the (P1) and (C1) curves, are for the case where the load is uniformly distributed among all cache-modules. The two other curves (P2) and (C2) depict the performances under the linearly distributed load.

As Figure 10 demonstrates, the "CC/PTB" architecture, in both its implementations, completely dominates the "Store-Controller" in three out of the four technology scenarios. Only in the first scenario (n = 2) does the "Store-Controller" show a performance advantage and only for high READ rates. The remaining part of this section will be devoted to an analysis of the different factors affecting the performance patterns demonstrated in Figure 10. Of particular help in conducting this analysis is the information of Figure 11 depicting the system bottlenecks in all the cases tested.

There are two basic patterns that deserve separate analysis. The distinctive pattern of n=2, and the pattern common among the three other scenarios, n=4, 6, and 8. To analyze the latter we will arbitrarily pick the case of n=6 as our analysis vehicle.

## V.5.1. Case of n = 2

To many, the most surprising aspect of these results will be the almost identical shapes of the "OPT" curve, and the "Store-Controller" curve (S1). To understand why this is so, we first note from Figure 11 that in both models LBUS2 (i.e., the local bus of level 2) is the bottleneck. Now, the difference between the "Store-Controller" model

	n = 2	n = 4	n = 6	n = 8
Program	% READS 70 80 85 90	% READS 70   80   85   90	% READS 70   80   85   90	% READS 70   80   85   90
ОРТ	← LBUS2	<	SLC	
	LBUS2	<del></del>	SC	
STCR	LBUS2	SC		
	LBUS2 LBUS1	<del></del>	SLC	
CC/PTB/C	LBUS2 LBUS1	SLC C1*	SLC C <sub>1</sub> *	SLC C1"
	LBUS2 LBUS1		SLC	
CC/PTB/P	LBUS? LBUS1	<del></del>	SLC	

 ${}^{\star}C_{1}$  is the Cache-Module carrying 30 percent of the load

Figure (11)

and the "OPT" model lies in the overhead necessary to handle the cache-consistency problem. All this overhead (in the "Store-Controller" model) is in the form of additional operations which all take place at the cache level. Thus, while removing this overhead (in the "OPT" model) will necessarily decrease the load on the cache level, it will not decrease the load on level 2, and in particular on LBUS2. Thus, since LBUS2 is the bottleneck in both cases, and since the load on it (by an "average" transaction) remains the same, the performances in both are very similar. For the same reasons, the other five models, namely, (S2), (P1), (P2), (C1), and (C2), have performances close to that of "OPT" for % READs, = 80 (i.e., they all have LBUS2 as the system bottleneck).

The fact that LBUS2 is the bottleneck is itself, by the way, an interesting finding. With a hit ratio of 0.90 for READ requests and 1.00 for WRITE requests most of the "action" is clearly done at the cache level. It would, therefore, seem that LBUS1 must be saturated before LBUS2. The answer goes back to the parameter values of section V.2. Notice that the transaction size for level 2 (64 bytes) is eight times larger than that for level 1 (8 bytes). This means that a single transmission on LBUS2 will be approximately eight times longer in time than a single transmission on LBUS1. Thus, although the absolute number of transmissions on LBUS2 is less than that on LBUS1, the utilization of LBUS2 in this case is higher.

Another interesting observation relates to curve (S2) of the "Store Controller." Curve (C2) is always below curve (S1) because in (S2) the "Store Controller" (SC) is the system bottleneck with a

saturation point lower than that of LBUS2. The reason this happens is that the high degree of data sharing between the caches (and which is "orchestrated" by the Store-Controller) means a higher utilization of the Store-Controller by the "average" READ/WRITE request. Notice also that the two curves (S1) and (S2) diverge as the % of READs increases. This is merely a reflection of the pattern by which the READ and WRITE operations utilize the two respective bottlenecks, LBUS2 and SC. (Straightforward analytic calculations would demonstrate that the effective capacity of LBUS2 increases faster than does that of SC as the % of READs increases.)

Next let us turn our attention to the "CC/PTB" results. Notice first the deflections in curves (P1), (P2), (C1), and (C2). This happens because at approximately 80 % READs LBUS1 (and not LBUS2) becomes the system bottleneck in the four cases. However, even though LBUS1 is the bottleneck for both "CC/PTB/P" and "CC/PTB/C", "CC/PTB/P" clearly dominates. This simply is because an "average" READ/WRITE request utilizes LBUS1 less often in "CC/PTB/P" than it does in "CC/PTB/C." For example, in CC/PTB/P when a requested data item is found by the processor not to be in the cache level, a request is sent to main memory via LBUS1. In CC/PTB/C, on the other hand, a CPU request must first go to a cache-module (through LBUS1), only to be found unavailable, and then forwarded by the cache-module to main memory through LBUS1 again.

Notice finally the negligible effect that the load distribution on the cache-modules has on performance (i.e., curves (C1) and (C2) are similar, as well as curves (P1) and (P2)). The reasons for this are completely analogous to the ones explaining the close resemblance between the "OPT" curve and curve (S1). In other words, changing the load distribution does not affect the utilization of LBUS1. As long as no new bottleneck develops because of the change in the load distribution, LBUS1 will remain to be the bottleneck, and thus maintain approximately the same throughput. The utilization of the cache-module that carries the highest load under the linear load distribution for both "CC/PTB/P" and "CC/PTB/C" turns out never to exceed 60 %, which is far below the 100 % mark that has to be approached before it would replace LBUS1 as the system bottleneck. This, in a sense, is very comforting to know. It shows that the behavior of the models is, to a large extent, insensitive to the shape of the load distribution on the cache-modules that we used.

## V.5.2. Case of n = 6

Most of the ideas of the above discussion are applicable to the case of n=6. For example, "OPT," "CC/PTB/P".s (Pl) and (P2), and "CC/PTB/C".s (Cl) all have almost identical performances because they all have the same bottleneck, namely, the Storage Level Controller (SLC).

Notice, on the other hand, that because the bus is now <u>relatively</u> faster in comparison to the other system components, and in particular to the Store-Controller (SC), LBUS2 ceases to be the bottleneck in the two "Store-Controller" models. Instead, SC is now the bottleneck, and the degradation in performance is quite evident. However, notice that even though SC is the bottleneck for both (S1) and (S2), the

"average" READ/WRITE request in (S2) (where there is 50 % data sharing) requires more "services" from the "Store-Controller" than does an "average" READ/WRITE request in (S1). (When data sharing exists between the cache-modules, the SC has, for example, to invalidate redundant copies when a data item is modified.)

The only remaining result that deserves some explanation, is the deflection exhibited in curve (C2) of "CC/PTB/C" at % READs = 80. The reason for this behavior is that somewhere between % READs = 80 and % READs = 85 the most heavily loaded cache-module (i.e., the one carrying 30 % of the load) replaces SLC as the system.s bottleneck. (See Figure 11). Notice that this does not happen in "CC/PTB/P.s" curve (P2) even though the same linear load distribution is used. The reason for this is because, even though, the cache-modules in both cases are subjected to the same load distribution, they are not subjected to the same load. This, of course, is because the READ/WRITE operations in the "CC/PTB/P" implementation use the cache-modules less often.

## V.6. Conclusion

The above results clearly indicate that no one approach dominates over <u>all</u> four technology scenarios. Technology, therefore, must remain as an element of some uncertainty.

It is important to realize, though, that what is really important in our technology forecasts is not the absolute values of the different speeds, but rather the <u>relative</u> values for the different hardware components. And the most important such value is the relative speed of the bus vis-a-vis the processing components that use it. Our results clearly demonstrate that the faster the bus is relative to everything else, the more appealling the "CC/PTB" approach becomes. More specifically, when the bus is four times as fast as the cache or faster (i.e.,  $n \rightleftharpoons 4$ ), the "CC/PTB" approach provides a 20 to 60 % performance advantage over the "Store-Controller" approach.

But, what perhaps is the most interesting finding, is the fact that for three out of the four technology scenarios (with  $r_i \Leftarrow = 4$ ) the performance of our "CC/PTB" scheme is very close to that of "OPT."

Notice that in the above statements no attempt was made to single out any of the two different implementation schemes of the "CC/PTB" approach. One of the interesting findings in the simulation results is that the performances of both schemes are very close indeed. Our own intuition was that the "CC/PTB/P" implementation would display a performance advantage. It was clear, that by utilizing "fast" processors that would overlap address translation with arbitrating for the bus, the utilizations of the bus and the cache-modules would decrease. Although the utilizations were indeed lower, this did not materialize into the higher performance we anticipated. The reason: the execution of the INFOPLEX storage hierarchy operations and storage mechanisms was such that the storage level controller (SIC), whose utilization is independent of the "CC/PTB" scheme used, would, in most

cases, be the first component to saturate. The only exception to this is when, under the linear load distribution case, the most heavily utilized cache-module developed into the system.s bottleneck. In such a case the higher performance potential of the "CC/PTB/P" scheme was indeed realized.

#### VI. CONCLUSION

This paper is a report on an ongoing research effort at the M.I.T. Sloan School of Management to study the multicache-consistency problem in highly parallel multi-processor computer systems.

There are three basic approaches proposed in the literature to handle the multicache-consistency problem: The "Broadcasting" approach, the "Store-Controller" approach, and the "Multics" approach. However, serious drawbacks could be identified in each. A new approach called the "CC/PTB" was, therefore, developed. It attempts to minimize performance degradation by minimizing the overhead of maintaining cache-consistency.

The "CC/PTB" approach was implemented in the INFOPLEX storage hierarchy and evaluated using simulation modeling. The results are very favorable.

This work opens up many areas for further investigation. No mention was made in this paper, for example, of the replacement algorithms at the cache level. It would be interesting to find out the value of implementing a "sophisticated" algorithm such as the "Least Recently Used" (LRU) algorithm (or versions of it) as opposed to a naive algorithm e.g., random replacement that requires much less hardware overhead.

We have assumed, as is common in the literature, that all WRITE operations for a particular data item are preceded by READ operations. Relaxing this constraint, and developing efficient algorithms to exploit both this relaxation and the architecture of the storage hierarchy is definately worth investigating.

And finally, the "CC/PTB" architecture should be exploited in the development of algorithms that would improve the reliability of the data storage hierarchy. The automatic data repair algorithms, for example, are particularly interesting and promising.

#### BIBLICGRAPHY

- 1. Censier, Lucien M. and Feautrier, Paul. "A New Solution to Coherence Problems in Multicache Systems," IEEE Trans. on Computers, Vol. C-27, No. 12, (Dec., 1978), 1112-1118.
- 2. Ferrari, D. Computer Systems Performance Evaluation. Maglewood Cliffs, New Jersey: Prentice-Hall, Inc., 1978.
- 3. Goodrich II, E. R. "A Distributed Operating System for a Central Shared Bus Multiprocessor System." Unpublished Master Thesis, M.I.T., 1980.
- 4. Greenberg, B. An internal Honeywell memorandum, 1978.
- 5. Kaplan, K. R. and Winder, R. O. "Cache-based Computer Systems,"
  Computer, (March, 1973), 31-36.
- 6. Lam, C. Y. "Simulation Studies of the DSH-II Data Storage
  Hierarchy System." M.I.T. Sloan School Internal Report No.
  M010-7506-04, 1979a.
- 7. Lam, C. Y. "Data Storage Hierarchy Systems for Data Base Computers." Unpublished Ph.D. Thesis, M.I.T., 1979b.
- 8. Madrick, S. E. "INFOPLEX Hierarchical Decomposition of a Large Information Management System Using a Microprocessor Complex," AFTPS Proc., Vol. 44, 1975, 581-587.
- 9. Machick, S. E. "The INFOPLEX Database Computer: Concepts and Directions," Proc. IEEE Computer Conference, Feb. 26, 1979, 168-176.
- 10. Matick, R. Computer Storage Systems & Technology. New York: John Wiley & Sons, 1977.
- 11. Tang, C. K. "Cache System Design in the Tightly Coupled Multiprocessor System," AFTPS Proc., Vol. 45, 1976, 746-753.
- 12. Toong, H. D.; Strommen, S. O.; and Goodrich II, E. R. "A General Multi-Microprocessor Interconnection Mechanism for Non-Numeric Processing," Proc. of the Fifth Workshop on Computer Architecture for Non-Numeric Processing, 1980, 115-123.

APPENDIX (I) : The "OPT" Program

...

CONVERSATIONAL MONITOR SYSTEM

Š VS1 JGB FILE: JPT

//TAR1 JOB TAR, // PROFILE='DEFER' // TIME=5

/\*PASS\*ORD SCUBA /GPSS PRGC /C EXEC PGM=DAG01,TIME=&TLIMIT

//Sreplib DD DSN=POTLUCK.LIBRARY.GPSS.LOAD.DISP=SHR //DOUTEUT DD SYSDUT=PROFILE=PRINT.DCB=BLKSIZE=931 //DINTERD DD UNIT=SCRATCH.SPACT=(CYL.(1,1)).DCB=BLKSIZE=1880 //DSYMIAB DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=7112 //DREPISEN DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=800 //DINTAGRA DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB-BLKSIZE=800

//STEP: EXEC GPSS.PARM=C,TLIMIT=9 // PEND

//DINPUTI DD \*
REALLOCATE FUN.5.QUE.10.FAC.50.BVR,200.BLG,2000,VAR,50
REALLOCATE FSV,50,HSV,10,COM,40000

OPT 

PARW USAGE

CPU ID + TXW ARRIVAL TIME+ TXW COMPL TIME \*

TAN EXEC TIME

DUMBY

WODEL COMPONENTS

CACHES: B11,...D15
LEYEL CONTRL: K1,K2
REG PPOCS: R2 BUSES: GBUS, LBUS1

STORIGE: D21 STORIGE: R1, R0 STORIGE: S1, S0 STORIGE: T1, T0 STORIGE: A1, AC STORIGE: 01, G0

MODEL PARAMETERS

PAGE 001

PAGE 002

CONVERSATIONAL MONITOR SYSTEM

4

VS1 .05

FILE: DPT

PAGE 003

CONVERSATIONAL MONITOR SYSTEM 5\$RIR2,10/5\$SIR2,10/5\$TIR2,10/5\$AIR2,10/5\$DIR2,10 S\$ROK1,10/S\$SOK1,10/S\$TIK1,10/S\$AIK1,10/S\$OOK1,10 \$\$RIK2.10/5\$SIK2.10/S\$TIK2.10/S\$AIK2.10/55GIK2.10 \$\$ROK2.10/5\$5OK2.10/5\$TOK2.10/5\$ADK2.10/55GOK2.10 SSRIDIT, 10/5\$SIDIT, 10/5\$TIDIT, 10/5\$AIDIT, 10 SSRICIZ, 10/5\$SIDIZ, 10/5\$TIDIZ, 10/5\$AIDIZ, 10 SSRIDIZ, 10/5\$SIDIZ, 10/5\$TIDIZ, 10/5\$AIDIZ, 10 SSRIDIZ, 10/5\$SIDIZ, 10/5\$TITT, 10/5\$AIDIZ, 10 SSRIDIZ, 10/5\$SIDIZ, 10/5\$TIDIZ, 10/5\$AIDIZ, 10 STD9AGE S\$RID21,10/5\$51021,10/5\$TID21,10 WICHA FUNCTION P1,05 1,4441/2,4412/3,44415 #1CH# FUNCTION P1,DS 1,###11/2,###12/3,###13/4,###14/5,###15 4 经共享分支收收 经运动转换 医电影电影 计电影电影 医共和外状体 电极电极系统 医自然性动物医动物 STOREGE FOR K2.K3.K4 STORAGE FOR REQ PROC VS1 JOB STORAGE FOR DEVICES BODIEAN VARIABLES STORAGE FOR L(1) CACHES STORISE FOR KI STORAGE STORAGE STORAGE STORAGE STORAGE STORAGE STORAGE STORAGE 140 FILE:

PAGE 004

1	•			
BY FIGHT BY AND LABY AND LABY BY FIRST BY FIRST BY FIGHT BY AND LABY BY FIRST BY FIR			***	
RTOM2 3-APT ABLE FNUSGBUS-SNFSTIKT  BY FCR	13	Y 133	* * ±5.2	
RTOM2 3.API ABLE FNUSGBUS*SNFSTIK1  BY FCR. 1  CMR! 3.451 13.E FNUSLBUS1*SNFSROKI CKCI 3.451 13.E FNUSLBUS1*SNFSROKI CKCI 3.451 13.E FNUSLBUS1*SNFSTID1 KDT11 3.451 13.E FNUSLBUS1*SNFSTID1 KDT11 3.451 13.E FNUSLBUS1*SNFSTID1 KDT13 3.451 13.E FNUSCBUS*SNFSTIX1 KKT21 8.451 13.E FNUSCBUS*SNFSTIX2 KKT22 8.451 13.E FNUSCBUS*SNFSTIX2 KKT21 8.451 13.E FNUSCBUS*SNFSTIX2 KKT21 8.451 13.E FNUSCBUS*SNFSTIX2 KKT21 8.451 13.E FNUSCBUS*SNFSTIX2 KKT22 8.451 13.E FNUSCBUS*SNFSTIX2 KKT21 8.451 13.E FNUSCBUS*SNFSTIX2 KKT21 8.451 13.E FNUSCBUS*SNFSTIX2 KKT22 8.451 13.E FNUSCBUS*SN		*****		
BY FOR LATE AND LEGUSIS SNESSONS  CART BLATLAGE FNUSTBUSIS SNESSONS  CKOT BLATLAGE FNUSTBUSIS SNESSONS  CKOT BLATLAGE FNUSTBUSIS SNESSONS  KNOT12 BLATLAGE FNUSTBUSIS SNESTED  KNOT2 BLATLAGE FNUSTBUSIS SNESTED  KNOT2 BLATLAGE FNUSTBUSIS SNESSINS  KKKT2 BLATLAGE FNUSTBUSS SNESSINS  KKKT2 BLATLAGE FNUSTBUSS SNESSINS  KKRT2 BLATLAGE FNUSTBUSS SNESSINS  KKRT2 BLATLAGE FNUSTBUSZ SNESSINS  KRT2 BLATLAGE FNUSTBUSZ SNESSINS  KRT3		ET 1017	STIK	
BY FCR LANGE BUST-SN FSCAKIONS BARNES BUST-SN FSCAKIONS BARNES BUST-SN FSCAKIONS BARNES BARNES BUST-SN FSCAKIONS BARNES B		ē	*	
CART B.ATT 19LE FNUSTBUST-SNESSOKY CKST B.ATT 19LE FNUSTBUST-SNESSOKY CKOTT B.ATT 19LE FNUSTBUST-SNESTON KDT11 B.ATT 19LE FNUSTBUST-SNESTON KDT12 B.ATT 19LE FNUSTBUST-SNESTID1 KDT13 B.ATT 19LE FNUSTBUST-SNESTID1 KDT14 B.ATT 19LE FNUSTBUST-SNESTID1 KDT15 B.ATT 19LE FNUSTBUST-SNESTID1 KDT15 B.ATT 19LE FNUSTBUST-SNESTIN1 KKT12 B.ATT 19LE FNUSTBUST-SNESTIR1 KKT12 B.ATT 19LE FNUSTBUST-SNESTIR1 KKT12 B.ATT 19LE FNUSTBUST-SNESTIR1 KKT21 B.ATT 19LE FNUSTBUST-SNESTIR1 KKT21 B.ATT 19LE FNUSTBUST-SNESTIR1 KKT21 B.ATT 19LE FNUSTBUST-SNESTIR2 KKT21 B.ATT 19LE FNUSTBUSZ-SNESTIR2	ť	•	• (	
CART 3.45 125 E FUSIBUST-SNESSOKT  CKOT 8.45 125 E FUSIBUST-SNESSOKT  CKOTT 8.45 125 E FUSIBUST-SNESTICT  KDTT1 8.45 125 E FUSIBUST-SNESTICT  KDTT1 8.45 125 E FUSIBUST-SNESTICT  KDTT3 8.45 125 E FUSIBUST-SNESTICT  KDT7 8.45 125 E FUSIBUST-SNESTICT  KDT7 8.45 125 E FUSIBUST-SNESTICT  KDT7 8.45 125 E FUSIBUST-SNESTICT  KDAT3 8.45 125 E FUSIBUST-SNESTICT  KDAT4 8.45 125 E FUSIBUST-SNESTICT  KCAT2 8.45 125 E FUSIBUST-SNESTICT  KKR12 8.45 125 E FUSIBUST-SNESTICT  KKR2 8.45 135 E FUSIBUST-SNESTICT  KR72 8.45 135 E FUSIBUST-SNESTICT  KR73 8.45 135 E FUSIBUST-SNESTICT  KR74 8.45 135 E FUSIBUST-SNESTICT  KR75 8.45 155 E FUSIBUST-SNESTICT  KR75	) ).	•	• •	
CART B. 45 1 13 LE FNUSLBUSTSN FSROKT  CKOT B. 45 1 45 LE FNUSLBUSTSN FSSOKT  CKOTT B. 45 LE FNUSLBUSTSN FSSOKT  KNOTT B. 45 LE FNUSLBUSTSN FSTIDT  KKKST B. 45 LE FNUSLBUSTSN FSTIDT  KKKST B. 45 LE FNUSCBUSTSN FSTIDT  KKKST B. 45 LE FNUSCBUSTSN FSTINT  KKKST B. 45 LE FNUSCBUSTSN FSTIKT  KKKST B. 45 LE FNUSCBUSTSN FSTIKT  KKRST B. 45 LE FNUSCBUSTSN FSTIKT  KKRS BVA3 ASLE FNUSCBUSTSN FSTIRT  KKRS BVA3 ASLE FNUSCBUSTSN FSTIRT  KRRZ BVA3 ASLE FNUSCBUSTSN FSTIRZ  KRRZ BVA	*****		* • • • • •	
DKS1 B.42145_E FNUSLBUS1*SNFSSOKI  DK011 B.42145_E FNUSLBUS1*SNFSTID1  KD112 B.42145_E FNUSLBUS1*SNFSTID1  KD113 B.42145_E FNUSLBUS1*SNFSTID1  KD114 B.42145_E FNUSLBUS1*SNFSTID1  KD414 B.42145_E FNUSLBUS1*SNFSTID1  KD414 B.42145_E FNUSLBUS1*SNFSTID1  KD415 B.42145_E FNUSLBUS1*SNFSTID1  KD414 B.42145_E FNUSLBUS1*SNFSTID1  KD415 B.42145_E FNUSLBUS1*SNFSTID1  KD415 B.42145_E FNUSGBUS*SNFSSIX2  KKK12 B.42145_E FNUSGBUS*SNFSSIX2  KKK12 B.42145_E FNUSGBUS*SNFSSIX2  KKK21 B.42145_E FNUSGBUS*SNFSSIX2  KKA21 B.42145_E FNUSLBUSZ*SNFSSIX2  KKA21 B.43145_E FNUSLBUSZ*SNFSSIX2  KKA21 B.44145_E FNUSLBUSZ*SNFSSIX2  KKA21	_	.62 . 23	USLBUS1 * SNFSR	
CKOI BAATASE FNUSLBUSI-SNESCAKI KDTI1 BAATASE FNUSLBUSI-SNESTICI KDTI1 BAATASE FNUSLBUSI-SNESTICI KDT13 BAATASE FNUSLBUSI-SNESTICI KDT13 BAATASE FNUSLBUSI-SNESTICI KDA13 BAATASE FNUSLBUSI-SNESTICI KDA13 BAATASE FNUSLBUSI-SNESTICI KDA13 BAATASE FNUSLBUSI-SNESTICI KDA14 BAATASE FNUSLBUSI-SNESTICI KDA14 BAATASE FNUSCBUSI-SNESTICI KAATA BAATASE FNUSCBUS-SNESSIKZ KKK12 BAATASE FNUSCBUS-SNESSIKZ KKK12 BAATASE FNUSCBUS-SNESSIKZ KKAZ1 BAATASE FNUSCBUS-SNESTIKI KKAZ1 BAATASE FNUSCBUS-SNESTIKI KRAZ2 BAATASE FNUSCBUS-SNESTIRZ KRAZ2 BAATASE FNUSLBUSZ-SNESTIRZ	-	- me :	NUSTBUST+SNESSO	
KUDITI BANKA ACLE FNUSLBUSI-SNI-SILDI KUDITI BANKA AGLE FNUSLBUSI-SNI-SILDI KUDITI BANKA AGLE FNUSLBUSI-SNI-SILDI KUDITI BANKA AGLE FNUSLBUSI-SNI-SILDI KUDATA BANKA AGLE FNUSLBUSI-SNI-SILDI KUDATA BANKA AGLE FNUSLBUSI-SNI-SILDI KUDATA BANKA AGLE FNUSLBUSI-SNI-SILDI KUDATA BANKA AGLE FNUSLBUSI-SNI-SILDI KKAZI BANKA AGLE FNUSGBUS-SNI-SILXI KKAZI BANKA AGLE FNUSGBUS-SNI-SILXI KKAZI BANKA AGLE FNUSGBUS-SNI-SILXI KRAZI BANKA AGLE FNUSLBUSZ-SNI-SILXI KRAZI BANKA AGLE FNUSLBUSZ-SNI-SILRI		10 to	NOSI * SN PSCOK 1	
KD713 B.ASTASE FNUSLBUS1*SN FSTID1 KD713 B.ASTASE FNUSLBUS1*SN FSTID1 KD713 B.ASTASE FNUSLBUS1*SN FSTID1 KD713 B.ASTASE FNUSLBUS1*SN FSTID1 KD713 B.ASTASE FNUSLBUS1*SN FSAID1 KD713 B.ASTASE FNUSLBUS1*SN FSAID1 KD713 B.ASTASE FNUSLBUS1*SN FSAID1 KD713 B.ASTASE FNUSGBUS*SN FSAID1 KKR12 B.ASTASE FNUSGBUS*SN FSIK2 KKR21 B.ASTASE FNUSGBUS*SN FSIK1 KKA21 B.ASTASE FNUSGBUS*SN FSIK2 KKA21 B.ASTASE FNUSGBUS*SN FSIK2 KKA21 B.ASTASE FNUSGBUSZ*SN FSIK2 KKA21 B.ASTASE FNUSLBUSZ*SN FSIK3 KKA31 B.ASTASE FNUSLBUSZ*KN FSIK3 KKA31 B.A	- :		EBUS1 * SN F ST 101	
KD714 BJA21A3LE FNUSLBUS1+SNFSTID1 KD413 BJA21A3LE FNUSLBUS1+SNFSTID1 KD414 BJA21A3LE FNUSLBUS1+SNFSAID1 KD414 BJA21A3LE FNUSLBUS1+SNFSAID1 KD414 BJA21A3LE FNUSLBUS1+SNFSAID1 KD414 BJA21A3LE FNUSLBUS1+SNFSAID1 KD415 BJA21A3LE FNUSLBUS1+SNFSAID1 KKR12 BJA21A3LE FNUSGBUS+SNFSSIK2 KKG12 BJA21A3LE FNUSGBUS+SNFSSIK2 KKG12 BJA21A3LE FNUSGBUS+SNFSSIK2 KKAZ1 BJA21A3LE FNUSGBUS+SNFSSIK1 KKAZ1 BJA21A3LE FNUSGBUS+SNFSSIK1 KKAZ1 BJA21A3LE FNUSGBUS+SNFSSIK1 KKAZ2 BJA21A3LE FNUSGBUS+SNFSSIK1 KKAZ2 BJA21A3LE FNUSLBUSZ+SNFSIRRZ KRZ2 BJA21A3LE FNUSLBUSZ+SNFSIRRZ KRZ2 BJA21A3LE FNUSLBUSZ+SNFSIRZ RORZ1 BJA3LE FNUSLBUSZ+SNFSIRZ RORZ1 BJA3LE FNUSLBUSZ+SNFSIRZ RORZ1 BJA3LE FNUSLBUSZ+SNFSIRZ	٠ <u>٣</u>		EUS1+SNFSTID1	
KD115 BJA91 AGLE FNUSLBUS1+SNFSTID1 KD413 BJA91 AGLE FNUSLBUS1+SNFSAID1 KD414 BJA91 AGLE FNUSLBUS1+SNFSAID1 KD414 BJA91 AGLE FNUSLBUS1+SNFSAID1 KD414 BJA91 AGLE FNUSLBUS1+SNFSAID1 KD415 BJA91 AGLE FNUSLBUS1+SNFSAID1 KKR12 BJA91 AGLE FNUSGBUS+SNFSSIK2 KKK12 BJA91 AGLE FNUSGBUS+SNFSSIK2 KKR21 BJA91 AGLE FNUSGBUS+SNFSSIK2 KKR21 BJA91 AGLE FNUSGBUS+SNFSSIK2 KKR21 BJA91 AGLE FNUSGBUS+SNFSSIK2 KRR2 BJA91 AGLE FNUSLBUS2+SNFSAIR2	4	TOT TOTA	LBUS1+SNF\$TID1	
MUNIT BRAZINE FUNCEUSISMINSTRUCT RUDATA BRAZINE FUNCEUSISMINSTRUCT RUDATA BRAZINE FUNCEUSISMINSTRUCT RUDATA BRAZINE FUNCEUSISMINSTRUCT RUDATA BRAZINE FUNCEUSISMINSTRUCT RKRIZ BRAZINE FUNCEUSISMINSTRUCT RKRIZ BRAZINE FUNCEUSISMINSTRUCT RKRZZ BRAZINE FUNCEUSISMINSTRUCT RKRZZ BRAZINE FUNCEUSISMINSTRUCT RKRZZ BRAZINE FUNCEUSZSNINSTRUCT RKRZZ BRAZINE FUNCEUSZSNINSTRUCT RKRZ BRAZINE FUNCEUSZSSNINSTRUCT RKRZ BRAZINE FUNCEUSZSNINSTRUCT RKRZ BRAZINE FUNCEUSZSSNINSTRUCT RKRZ BRAZINE FUNCEUSZSNINSTRUCT RKRZ BRAZINE FUNCEUSZSSNINSTRUCT RKRZ BRAZINE FUNCEUSZNINSTRUCT RKRZ BRAZINE FUNCEUSZSSNINSTRUCT RKRZ BRAZINE FUNCEUSZNINSTRUCT RKRZ BRAZINE FUNCEUS RKRZ BRAZINE FUNCEUS RKRZ BRAZINE FUNCEUS RKRZ RKRZ BRAZINE FUNCEUS RKRZ RKRZ RKRZ RKRZ RKRZ RKRZ RKRZ RKR	n i	COT I SE	LBUS1*SNFSTID1	
### ### ### ### ### ### ### ### ### ##	- (		EBUST # SNF SALDT	
KDA14 B.ASTANE FNSEEUST-SNFSAIDT  ***********************************	4 (7)		NUSCEUS1 *SNFSAID1	
MOAIS SITRIAGE FNUSCOUST-SNFSAIDI  **********************************	4	104 104	NUSLEUS1*SNFSAID1	
BY FOR INTER LEVEL COM*  KKR12 BLAFLABLE FNUSGBUS*SNFSSIKZ KKC312 BLAFLABLE FNUSGBUS*SNFSSIKZ KKC312 BLAFLABLE FNUSGBUS*SNFSSIKZ KKC312 BLAFLABLE FNUSGBUS*SNFSSIKZ KKA21 BLAFLABLE FNUSGBUS*SNFSAIK1  KRA21 BLAFLABLE FNUSGBUSZ*SNFSAIK1  KRA22 BVAFLABLE FNUSLBUSZ*SNFSAIR KRA22 BVAFLABLE FNUSLBUSZ*SNFSSIR KRA22 BVAFLABLE FNUSLBUSZ*SNFSSIR KRA22 BVAFLABLE FNUSLBUSZ*SNFSAIR KRA22 BVAFLABLE FNUSLBUSZ*SNFSAIR KRA22 BVAFLABLE FNUSLBUSZ*SNFSAIR KRA22 BVAFLABLE FNUSLBUSZ*SNFSAIR	A 15	(I) of (I	NUSLBUS1 *SNFSAID1	
FOR INTER LEVEL COM**  2 Bitslade FNUSGBUS*SNFSRIK2 2 Bitslade FNUSGBUS*SNFSRIK2 2 Bitslade FNUSGBUS*SNFSRIK1 1 Bitslade FNUSGBUS*SNFSRIK1 1 Bitslade FNUSGBUS*SNFSRIR 1 Bitslade FNUSGBUSZ*SNFSRIR	•	*****	*	
FOR INTER LEVEL COM*  2 B.AT. ABLE FNUSGBUS*SNFSRIK2 2 B.AT. ABLE FNUSGBUS*SNFSSIK2 2 B.AT. ABLE FNUSGBUS*SNFSSIK2 1 B.AT. ABLE FNUSGBUS*SNFSTIK1 1 B.AT. ABLE FNUSGBUS*SNFSTIK1 1 B.AT. ABLE FNUSGBUS*SNFSTIK 1 B.AT. ABLE FNUSGBUSZ*SNFSTIK 1 B.AT. AB	•			
BYARIABLE FNUSGBUS*SNFSRIK2 BYARIABLE FNUSGBUS*SNFSSIK2 BYARIABLE FNUSGBUS*SNFSTIK1 BYARIABLE FNUSGBUS*SNFSAIK1 BYARIABLE FNUSLBUSZ*SNFSAIR BYARIABLE FNUSLBUSZ*SNFSAIR BYARIABLE FNUSLBUSZ*SNFSAIR BYARIABLE FNUSLBUSZ*SNFSAIR BYARIABLE FNUSLBUSZ*SNFSAIR BYARIABLE FNUSLBUSZ*SNFSAIR	5	IN ER	EL COM	
2 B.ARIABLE FNUSGBUS-SNFSRIKZ 2 B.ARIABLE FNUSGBUS-SNFSSIKZ 2 B.ARIABLE FNUSGBUS-SNFSTIKI 1 B.ARIABLE FNUSGBUS-SNFSAIKI 1 B.ARIABLE FNUSGBUS-SNFSAIK 1 B.ARIABLE FNUSGBUSZ-SNFSAIK 1 BVARIABLE FNUSGBUSZ-SNFSAIK 1 BVARIABLE FNUSGBUSZ-SNFSAIK			•	
2 B.ARTARE FNUSGOUS-SNFSRIKZ 2 B.ARTARE FNUSGOUS-SNFSSIKZ 2 B.ARTARE FNUSGOUS-SNFSIKI 1 B.ARTARE FNUSGOUS-SNFSTIKI 1 B.ARTARE FNUSCOUS-SNFSRIR	***		***	·
2 BYARIARE FRUSGBUS*SNFSSIKZ 2 BYARIARE FRUSGBUS*SNFSSIKZ 1 BYARIARE FRUSGBUS*SNFSTIKI 1 BYARIARE FRUSGBUS*SNFSAIKI  BYARIARE FRUSGBUSZ*SNFSRIR  BYARIARE FRUSGBUSZ*SNFSRIR  BYARIARE FRUSGBUSZ*SNFSRIR  BYARIARE FRUSGBUSZ*SNFSRIR  BYARIARE FRUSGBUSZ*SNFSRIR  BYARIARE FRUSGBUSZ*SNFSRIR	2	- HT 1 24 .	NUSGEUS+SNFSRIK	•
2 ByA3; A3LE FNUSGBUS*SNFSSOKZ 1 ByA3; A3LE FNUSGBUS*SNFSTIK1 1 ByA3; A3LE FNUSGBUS*SNFSAIK1 1 ByA3; A3LE FNUSGBUSZ*SNFSAIR 1 BYA3; A3LE FNUSGBUSZ*SNFSSIR	12	TET 1871	NUSGBUS*SNFSSIK	
1 B.ASTAGLE FNUSGGUS*SNFSTIKA 1 B.ASTAGLE FNUSGGUS*SNFSAIKA 1 B.ASTAGLE FNUSLGUSZ*SNFSAIKA 1 B.ASTAGLE FNUSLGUSZ*SNFSAIR	2	764 2 43"	NOSCENS+SNE & SOK	
BAARTABLE FNUSGBUS*SNFSAIKT  FOF L(2) OPS  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR	7	1841247	NUSGBUS * SNF \$ 1 IK	
FOR L(2) OPS **  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR  BVARIABLE FNUSLBUSZ*SNFSRIR	A21	. I I I I I I I I I I I I I I I I I I I	NUSGBUS+SNF SAIK	
FOR L(2) DPS *  BVARIABLE FNUSLBUSZ*SNFSRIR BVARIABLE FNUSLBUSZ*SNFSRIR BVARIABLE FNUSLBUSZ*SNFSRIR BVARIABLE FNUSLBUSZ*SNFSRIP BVARIABLE FNUSLBUSZ*SNFSRIP	****	*****	*	
FUR L(2) DPS *  BVARIABLE FNUSLBUSZ+SNFSRIR BVARIABLE FNUSLBUSZ-SNFSSIR BVARIABLE FNUSLBUSZ-SNFSSIR BVARIABLE FNUSLBUSZ-SNFSSIR BVARIABLE FNUSLBUSZ-SNFSSIR			•	
BVARIABLE FNUSLBUSZ+SNYSRIR BVARIABLE FNUSLBUSZ+SNFSIR BVARIABLE FNUSLBUSZ-SNFSIR BVARIABLE FNUSLBUSZ-SNFSIR BVARIABLE FNUSLBUSZ+SNFSUIR BVARIABLE FNUSLBUSZ+SNFSUIR	•	L(2) 3P	*	
BYARIABLE FNUSLBUSZ*SN*SRIR BYARIABLE FNUSLBUSZ*SNFSSIR BYARIABLE FNUSLBUSZ*SNFSTIR BYARIABLE FNUSLBUSZ*SNFSAIR BYARIABLE FNUSLBUSZ*SNFSSIR			* * *	
BVAPIABLE ENUSCHDUSZ+SMFSSIR BVAPIABLE FNUSCHUSZ+SMFSTIR BVAPIABLE FNUSCHUSZ+SMFSTIR BVAPIABLE FNUSCHUSZ+SMFSDIR 1 BVARIABLE FNUSCHUSZ+SMFSDIR		E4 1 647	81 BUS2*SN* SPIR	
BVARIABLE FNUSLBUSZ*SNFSTIR BVARIABLE FNUSLBUSZ*SNFSAIR BVARIABLE FNUSLBUSZ*SNFSDIR 1 BVARIABLE FNUSLBUSZ*SNFSRID		Cr cy.	SLBUS2*SNFSSIR	
BEARIAGEE FNUSLBUSZ-SNESAIR BVARIAGEE FNUSLBUSZ-SNESOIR 1 BVARIAGEE FNUSLBUSZ-SNESRID		ET ITA	SLBUS2=SNFSTIR	
BVATIAGLE FAUSTGUSZ*SNFSGIR 1 BVATIAGLE FNUSTBUSZ*SNFSRID		ARIAE	SLBUSZ+SNFSAIR	
1 BVARIABLE FNUSLBUS2*SNFSRID		VATI AS	SLBUSZ-SNFSOIR	
	_	PA 1 2 4 7	CACCOLLO COLLEGIO	

CONVERSATIONAL MONITOR SYSTEM

VS1 J08

BVARIABLE BVARIABLE BVARIABLE BVARIABLE BVARIABLE BVARIABLE

HAGRO -USE

**n** 

USE

\*\*\*\*\*\*

MACRO - FINI

44 FILE: 3PT

RDT21 DKS2 DKS2 DK12 DK13 RRR2 RRR2 RRR2 RRR2

FNUSLBUS2\*SNF\$TID21 FNUSLBUS2\*SNF\$TEXE FNUSLBUS2\*SNF\$TEXE FNUSLBUS2\*SNF\$ADKZ FNUSLBUS2\*SNF\$ADKZ FNUSLBUS2\*SNF\$ADKZ FNUSLBUS2\*SNF\$ADKZ FNUSLBUS2\*SNF\$ADKZ

STARTMACRO SELZE AUVANCE ASSIGN PELEASE ENDMACRO

WACRO - SEND

TRANSIT TIME BY FOR SEND OP

STARTMACRO TEST E ENTER SEIZE ADVANCE ASSIAN RELEASE LEAVE ENDMACRO SEND

MACROS

\*

USAGE TIME

BEGIN SIMPLATION

SIMULATE

3.5.7,9,11,13,15,17

SET HIGH P FOR NEW TXN ARRIVAL TIME CPU ID ... XSVAXMP.,,F GENERATE PRIJRITY Mark CPU1 STAR1

XSCOLS .XSNOCAD.WWW1.RRR1 .XSP[v1.NIN11.RIN11 ADVANCE TRANSFER TRANSFER RRR

DATA IS IN DATA CACHE .

PUT TXN IN READ REG BUFFER SEARCH AND READ CACHE FREE BUFFER RICHT XSRDEX1 DRP11.XSRDEX1 RICHT XSCGUS RIN1 ENTER
USE MACRO
LEAVE
ADVANCE
FINI MACRO

\*

. DATA IS NOT IN CACHE

A NEW TXN STAR:

TRANS FER

BC. 12V

FILE: DPT

\*

FINI

STATT MACRO SAVEVALUE SAVEVALUE SAVEVALUE SAVEVALUE SAVEVALUE SAVEVALUE ASSIGN ASSIGN ASSIGN ASSIGN ASSIGN ASSIGN

CPU +1

RIFULT

ASSIGN

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                                                                                                                                                                                                                                                                                                                                                                                   PUT TXN IN READ REG BUFFER
SEARCH AND READ CACHE
FREE BUFFER
                                                                                                                                                          PUT TAN IN ARITE REG BUFFER
                                                                                                                                                                                                                                                                                                                                    SET HIGH P FOR NEW TXN
ARRIVAL TIME
CPU ID
                                                                                      TO COMMON CODE FOR READ
                                           PUT IN READ REG BUFFER SEARCH DIRECTORY RESET PRICAITY
                                                                                                                                                                            WRITE DATA IN CACHE
                                                                                                                                                                                            0
SID11, SOK1, LBUS1, X $BEX8, BV $DKS1
                                                                       RID11, ROK1, LBUS1, X $BEX1, 3: SDKR1
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 A NEW TXN
                                                                                                                                                                                                                                                           A NEW TXN
                                                                                                                                                                                                                                                                                                                                                                    XSCBUS
.XSNREAD.WWW2.RRR2
.XSPIN1.NIN12.RIN12
                                                                                                                                                                                                                                                                                                                                                                                                                                                          RID12
DRP12,X$RDEX1
RID12
X$CBUS
                                                                                                                                                                                                                                                                                                                           .,XSMAXMP.,F
                                                                                                                                                                                DAP11, X$RDEX1
                                                 RID11
DRP11.XSREX
0
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    STAR2
                                                                                                                                                                                                                                                                                                                                                                                                                                         4
                                                                                                                                                                                                                                                             ,STAR1
                                                                                                                                                                                                                          1, COMW
                                                                                                                                                *******
                                                                                                                                                                                                                                                                                                                                                                                                                      DATA IS IN DATA CACHE
                                                                                                                              TO CACHE
                                                                                             , COMP
                                                                                                                                                                  $1011
          VS1 JOB
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    ADVANCE
WACRO
TRANS FER
                                                                                                                                                                                                                                                                                                                                                              ASSIGN
ADVANCE
TRANSFER
TRANSFER
                                                                                                                                                                                                                                                                                                                                     GENERATE
PRIORITY
                                                                                                                                                                                                                                                                 TRANS FER
                                                                                                                                                                                                    PRIORITY
WACRO
                                                                                                                                 WRITE REQUEST
                                                                                               TRANS FER
                                                                    PRIORITY
                                                                                                                                                                                                                                                                                                                                                                                                                                                                              LEAVE
                                                                                                                                                                                                                              SPLIT
                                                                                                                                                                                                                                                                                                                                                       VARK
                                                                                                                                                                  ENTER
                                                                                                                                                                                                                                                MACRO
                                                                                                                                                                                     MACRO
                                                                               MACRO
                                                    NIN11 ENTER
USE MACRO
                                                                                                                                                                                                                                                                                                    CP. #2
                                                                                                                                                                                                                                                                                                                                                                                                                                                             FIN:3
           FILE: 3PT
                                                                                                                                                                                                                                                                                                                                      CPUZ
STAR2
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              FIMI
                                                                                                                                                                                                                                                                                                                                                                                         4842
                                                                                                                                                                                                                                               FINI
                                                                                                                                                                                                                                                                                                                                                                                                                                                                       SE
                                                                                                                                                                    . . .
                                                                                                                                                                                                                SEXO
                                                                              SEND
```

FILE:	JPT	451,03	*	CONVERSATIONAL MONITOR SYSTEM
DATA	IS NOT	IN CACHE		
NIN12 USE SEND	ENTER WACFO PRIORIT	# 0 0 H	.XSREX .ROK1,LBUS1	PUT IN READ REQ BUFFER SEARCH DIRECTORY RESET PRIORITY , XSBEX1, BV\$DKR1
	TRANS FER	R , CGMR		TO COMMON CODE FOR READ
:.			* *	
· WRITE	E REQUES	7 73 CACHE*	# * *	
WW2	ENTER	SICIS		PUT TXN IN WRITE REQ BUFFER
USE	MACRO	04212	., X\$RDEX1	WRITE DATA IN CACHE
SEND	PRIORITY MACRO	r 0 SID12,	, SOK1 , L BUS1	RESET TXN PRIORITY ,X\$BEX8,BV\$DKS1
	SPLIT	1 , COMM		
FINI	MACRO			
	TRANSFER	R STAR2	7	A NEW TXN
Ü	0 73		••••	
CPU3 STAR3 RRR3	GENERATE PRIORITY MARK ASSIGN ADVANCE TRANSFER TRANSFER		9958AXMP,,,FSS 2013 11.3 C XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS XSCBUS	SET HIGH P FOR NEW TXN ARRIVAL TIME CPU IC 13
DATA	NI SI	DATA CACHE	* * * * *	
RIN13 USE	ENTER	IR CORD CORD CORD CORD CORD CORD CORD CORD	ALD13 DRP13. X\$RDEX1 RLD13	PUT TXN IN READ REQ BUFFER SEARCH AND READ CACHE FREE BUFFER

MONITOR SYSTEM										K W												
CONVERSATIONAL MONITOR SYSTEM	A NEW TXA				PUT IN READ REQ BUFFER SEARCH DIRECTORY RESET PRIORITY 'X\$BEX1,BV\$DKR1	TO COMMON CODE FOR READ				PUT TXN IN WRITE REQ BUFFER	WRITE DATA IN CACHE	RESET TXN PRIORITY	, X SBEXB, BVSDKS1							SET HIGH P FOR NEW TXN ARRIVAL TIME CPU ID	N14	
VS1 JOB A4	X\$CBUS ,STAR3	* * * * * * * * * * * * * * * * * * * *	CACHE .	***	PUT IN READ RE DRP13, X\$REX SEARCH DIRECTO O RESET PRIORITY RID13, ROK1, LBUS1, X\$BEX1, BV\$DKR1	, COMR	****	O CACHE.		SID13	DRF13,X\$RDEX1	•	SID13, SOM1, LBUS1, X\$BEX8, BV\$DKS1	1.COMM		STAR3	****	• • •		,,,X\$MAXMP,,,F 9 S 2 A 1,4 C X\$CBUS ,X\$NREAD,WWW4,RRR4	.XSPIN1,NIN14,RIN14	A CACHE .
FILE: 3PT VS	ADVANCE FINI MACRO TRANSFER	******	. DATA IS NOT IN	*****	NIN13 ENTER USE MACRO PRICRITY SEND MACRO	TRANS FER		* WRITE REQUEST TO	****	WWW3 EHTER	USE MACRO	TRIDRITY	SEND MACRO	SPLIT	FINI WACRO	TRANSFER	****	** 040	****	CPU4 GE'ERATE STAR4 PRICATIY MARK ASSIGN ADVANCE TRANSFER	BAR4 TRANSFER	• DATA IS IN DATA

JR SYSTEM						
CONVERSATIONAL MONITOR	PUT TAN IN READ REG BUFFER SEARCH AND READ CACHE FREE BUFFER A'NEM TAN	PUT IN READ REQ BUFFER SEARCH DIRECTORY RESET PRIORITY *XSBEX1, BVSDKR1 TO COMMON CODE FOR READ	PUT TXN IN WRITE REG BUFFER Write data in cache Reset txn priority	XSBEXB, BVSDKS1		SET HIGH P FOR NEW TXN ARRIVAL TIME CPU ID
VS1 JOB A4	xs x	AID14, X\$REX  2	SI014 DRP14, X\$RDEX1 0	SID14, SOK1, LBUS1, X SBEXB, BV SDKS1 1, COMW .STAR4 A NEW TXN		XSMAXMP,,,F SE 9 AR 2 AR 1,5 CP XSCBUS XSNREAD,WWWS,RRRS
FILE: 3PT VS	RIGIA ENTER RID14 USE WACRO DRP14. EAVE RID14 ADVANCE XSCBUS FINI WACRO TRANSFER STARA DATA IS NOT IN CACHE	10 ft 7 + G +	WING ENTER USE MACRO PRIORITY	SEND WACRD SPLIT FINI WACRD TRANSFER	CP 7 42	CPUS GENERATE STARS PRICAITY MARK ASSIGN ADVANCE TRANSFER

DATA I				
******	DATA IS IN DATA CACHE	CACHE		
RIN1S E USE L	ENTER MACRO LEAVE ADVANCE MACRO	RID15 DRP15, RID15 X\$CBUS	X\$RDEX1	PUT TXN IN READ REQ BUFFER SEARCH AND READ CACHE FREE BUFFER
* * *	TRANSFER	STAR5		A NEE TXN
NIN15 E USE N	ENTER MACRO PRIORITY MACRO	RID15 DRP15,XSREX 0 RID55,ROK1,	1. BUS 1	PUT 'N READ REG BUFFER SEARCH DIRECTORY RESET PAIDRIT' 'XSBEX1, BVSDKR1
	TRANS FER	, COMR	• •	TO COMMON CODE FOR READ
WRITE	WRITE REQUEST TO	TO CACHE	• • •	
MMMS	ENTER	\$1018		PUT TXN IN MUITE REG BUFFER
USE N	MACRO PRIDRITY	DRP15.	DRP15, XSRDEX1 0	WRITE DATA IN CACHE RESET TXN PRIORITY
SEND N	KACRD	SID15,	SOK1, LBUS!	SID15,SOK1,LBUS!,X\$BEX8,BV\$DK\$1
Fini W	SPLIT MACRO	1.COM	_	
_	TRANSFER	STAR5		A NEW TXN
W.DO	CO-HINCH CODE	FOR READ	REQUEST	

CONVERSATIONAL MONITOR SYSTEM RIR2, RID21, LBUS2, X SBEX1, BV\$RDR21 STORE DATA INTO L(1) AS RESULT OF A READ-THROUGH RID21, TOK2, LBUS2, X \$BEX8, BYSDKT2 ROK1. RIK2, GBUS, AJBEX1, BVSKKR12 RIK2, RIR2, LBUS2, X\$ BEX1, BYSKRR2 TOK2, TIK1, GBUS, X\$BEX8, BVSRTOK2 DRP21, XSDEX2 READ DATA IS FOUND IN L(2) KRP2,XSKEX RRP2,XSREX KRP1, XSKEX KRP2, XSKEX KRP1,X\$KEX \*\*\*\*\*\* 电电路电路电路 医多种动物 医甲基苯甲二甲基甲基苯酚 READ-THAQUGH TO L(1) 11.0 VS1 J03 DATA IS IN D21 RRR21 ASSIGN STOR1 ASSIGN MACRO WACRO CHOYA MACRO WACAC ひとつなる MACRO WACRO 06378 MACAG DECKN. FILE: JPT SEND CN35 SEND SEND SEND USE USE 155 USE LSE USE

```
PAGE 013
```

```
CONVERSATIONAL MONITOR SYSTEM
```

VS1 JOB

FILE: OPT

1. FNSWICHE 计分类设计分类 计电路电路 医非沙耳氏性医神经炎性 SPLIT TERMINATE

11.0 WWW11 ASSIGN

. RT STORE INTO D11

TIK1, TID11, LBUS1, X \$BEX8, BV\$KDT11 MACRO SEND

DRP11,X\$RPLR MACRC USE .X\$POV1, POV11,0VL11 TID11 X\$CBUS TRANSFER NOV11 LE.VE ADVANCE

MACRO

FINI

STAR1 TRANS FER 1,0VF11 X\$CBUS DVL11 SPLIT ADVANCE

FINI WACRO

.STAR1 TRANSFER DVF11 ASSIGN

TID11.00K1, LBUS1, X SBEX1, BV SDK01 SEND MACRO

,0VL1 TRANSFER

RT STORE INTO D12

11.0 WW12 ASSIGN

TIK1, TID12, LBUS1, X \$BEX6, BV\$KDT12 SEND MACRO

DRP12, XSRPLR MACRO

USE

.xspoy1,MOV12,OVL12 TID12 xscbus trans fer Leave Advance KOV12 1

MAC20 FINI STAR2 TRANSFER

CONVERSATIONAL MONIFOR SYSTEM

VS1 JOB A4

FILE: OPT

TRANSFER NOV14 LEAVE ADVANCE

MACRO

INI

TRANS FER

STAR4

DVL14 SPLIT ADVA

WALTO FINI

TRANSFER OVF14 ASSIGN

TID:4,00K1, LBUS1, X \$5EX1, BV\$DK01

TRANS FER

HT STORE INTO DIS

0,11

TIK1, TIU15, LBUS1, X \$8EX8, 8V\$KDT15

URP15, XSRPLR

WACRO

USE

STAR5 TRANSFER

TRANSFER OVF15 ASSIGN

SEND MACRO

.XSEGV1,NGV14,OVL14 TID14 XSCBUS

XSCBUS

.STAR4

CN3S

, **OV**L.

电分子电极 医经济性原体 医医院医院医院医院经验

MMM15 ASSIGN

SEND WACRO

.X\$PQV1,NOV15,OVL15 TID15 X\$CBUS TANNS FER NOVIS LEAVE

FINI

WACRO

1,0VF15 X\$CBUS DVL15 SPLIT ADVANCE

FINI MACRO

,STAR5

TID15, DGK1, LBUS1, X \$BEX1, BV\$DKG1

TRANSFER

.001

COK1.SIK2.C US.XSBEX1.BVSKKO12 DIKZ.CIRZ,LBUSZ,XSBEX1,BVSKROZ RRP2,XSREX KRP2,XSKEX KRP1, XSKEX 0172 11.0 \* MANDLE OVF FROM L(1) LEAVE TERMINATE OVL! ASSIGN MACRO MACRO MACPO MACRO MACRO SEND SEND USE USE SE

COMMON CODE FOR STURE-BEHIND

MACRO MACRO MACRO MACRO MACRO MACRO MACRO			
MACRO MACRO MACRO MACRO		IACRO	KRP1,XSKEX
MACRO MACRO MACRO MACRO MACRO		ACRO	SOK1,SIK2,GBUS,XS3EX3,BV\$KKS12
MACRO MACRO MACRO MACRO		MACRO	KRP2,XSKEX
MACRO MACRO MACRO		MACRO	SIKZ, SIRZ, LBUSZ, X\$ BEX8, BV\$KRSZ
MACRO		MACRO	RRP2,XSREX
MACRO		MACRO	SIR2, SI021, LBUS2, X \$3EX8, BV\$RDS21
		MACRO	DRP21, XSDEX2
MACRO	SEND	MACRO	SID21, ADK2, LBUS2, X\$BEX1, BV\$DKS2

11,0 ACK21 ASSIGN KRP2,XSKEX MACRO

USE

44

VS1 JOB

FILE: OPT

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                                                                                                                                                                                                                                                                                                                                                                                                   AIK1, AID13, LBUS1, X $BEX1, BV5KDA13
                                                                                                                                                                                                                                                                                                                      AIK1, AID12, LBUS1, XSBEX1, BV$KDA12
                                                                                                                                                                        AIK1, AID11, LBUS1, X$BEX1, BV$KDA11
                           AGK2, AIK1, GBUS, X$BEX1, BV$KKA21
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       DRP13, XSREX
                                                                                                                                                                                            DPP11, XSREX
                                                                                                                                                                                                                                                                                                                                        DRP12, XSREX
                                                KRP1, XSKEX
                                                                 1.FNSWICHA
                                                                                                                                                                                                                                                                                                                                                                                                                                 医心体性动脉炎 医医性性神经 医非神经性神经医学神经
                                                                                                                                                                                                                                              1. 电电子电子电子电子电子电子电子电子电子电子电子
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          AID13
                                                                                                                                                                                                                                                                                                                                                             A1012
                                                                                                                                                                                                                AID11
                                                                                                                                                                                                                                                                                                                                                                                                                                                  11.0
                                                                                                                                      0.1.
  VS1 J08
                                                                                                                                                                                                                                                                                                                                                                                                             ACK HANDLED BY D13
                                                                                                                                                                                                                                                              ACK HANCLED BY D12
                                                                                                                  ACK HANDLED BY D11
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           LEAVE
TERMINATE
                                                                   SPLIT
TERMI NATE
                                                                                                                                                                                                                                                                                                                                                              LEAVE
TERNINATE
                                                                                                                                                                                                                LEAVE
TERMI HATE
                                                                                                                                                                                                                                                                                                                                                                                                                                                  AAA13 ASSIGN
                                                                                                                                                       AAA11 ASSIGN
                                                                                                                                                                                                                                                                                                    AAA12 ASSIGN
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        MACRO
                                                                                                                                                                                            MACRO
                                                                                                                                                                                                                                                                                                                                        MACRO
                                                                                                                                                                                                                                                                                                                                                                                                                                                                      MACRO
                                                                                                                                                                                                                                                                                                                       SEND MACRO
                              MACRO
                                                MACRO
                                                                                                                                                                         SEND MACRO
  FILE: OPT
                                                                                                                                                                                                                                                                                                                                                                                                                                                                      SEND
                              SEND
                                                USE
                                                                                                                                                                                             USE
                                                                                                                                                                                                                                                                                                                                         USE
```

ACK MANDLED BY D14 ..

4 VS1 JOB FILE: OPT

AAA14 ASSIGN

AIK1, AID14, LEUS1, X SBEK!, BV\$KDA14 MACRO SEND

DRP14, XSREX MACRO

USE

LEAVE TERMINATE

A101A

. ACK HANDLED BY D15

11.0 AAA15 ASSIGN

AIK1, AID15, L9US1, XSBEX1, BV\$KDA15 SEND MACRO

DAP15, XSREX MACRO

USE

AID15 LEAVE TERMINATE

SIMULATION CONTROL

XSTIMER 1 GENERATE TERMINATE START END

SS END

APPENDIX (II): The "STCR" Program

VS1 JCB

4

//\*PASSWORD SCUBA

//C EXEC PGW=CAGGITIME=#TITTIT
//C EXEC PGW=CAGGITIME=#TITTIT
//STEPLIB DD DSN==DTLUCA.LIB=A=r.GPSS.LGAD.DISP=SHR
//DOUTPUT DD SYSCLI=PTITLE=PRINT.DCB=BLKSIZE=931
//DINTERD DD UNIT=SCPATCH.SPACE=CYL.(1,1)).DCB=BLKSIZE=1880
//DREPTGEN DD UNIT=SCPATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=800
//DINTERD DD UNIT=SCPATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=800
//PEDD
//STEP1 EXEC GPSS.PARW=C.TLIVIT=9
//DINPUTI DD \*\*
REALLDCATE FUN.S.QUE.10.FAC.50.8VR,200.BLD.2000.VAR,50
REALLCCATE FSV.50.MSV.10.CCW,40000

STCR 

ARRIVAL TIWE\* CCWPL TIWE \* TXN PARM USAGE

TXN EXEC TIME 2222

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* MODEL COMPONENTS # BUSES: GBUS, LBUS1,...
# CACHES: D11,...D15
# LEVEL CONTAL: K1,K2 \*
# REQ PPOCS: R1,R2 \*
# DEVICES: D21

STORAGE: RI, RO STORAGE: SI, SO STORAGE: TI, TC STORAGE: AI, AO STORAGE: OI, 30

\* MODEL PARAMETERS

\*\*\*\*\*\*\*\*\*\*

XSMODEL. INITIAL

MODEL STCR

PAGE 001

//TAR1 JOB TAR. // PROFILE='DEFER', // TIME=5

FILE: STCR

CONJERSATIONAL MONITOR SYSTEM	OG PER CPU	OF FINDING DATA IN L1		ME IN L1		FOR 8 BYTE BLOCK	FOR 64 BYTE BLOCK	SERVICE TIME BETWEEN CPU & CACHE	TIME	1#E	LG AND STORE IN L1	TIME OF CACHE		IT IN CACHE):	THER CACHE DOES NOT	INE AS A PRIVATE LINE	WAITING A LINE (& IN CACHE):	PROS. LINE MAS NOT JUST DECLARED	OTHER CACHE	WAITING (SLINE IN CACHE & LINE	NOT JUST DECLARED PRIVATE BY	::	OR MORE	OTHER CACHES DON'T SHARE THE LINE	BEHIND	0 #	SEARCHES DIREC	UPDT DIREC	DIREC	PDATES ITS DIREC	P	A SO IT WRITES	EC	DECR 'STORE-BEHIND ACK CNT'
CON VE	DEGREE OF WULTIPROG PER % READ 4EQ	CONDITIONAL PROB OF	IN L2	SEVICE SERVICE TIME IN LI		BUS SERVICE TIME	BUS SERVICE TIME	BUD SERVICE TIME	DIRECTORY LOOK UP TIME	CONTROLLER SERV TIME	TIME TO USE RPL ALG AND STORE IN	LCOKUP PLUS PEAD	SIMPLATION TIME	WHEN READING (SNOT IN CACHE):	% OF TIME ANOTHER	CONTAIN THE LINE	MHEN KAITING A LI	PROS. LINE NA	PRICATE BY ANDTHER CACHE	MHEN WAITING (SLI	NOT JUST DECL	ANOTHER CACHE):	PROB THAT ONE OR MORE	OTHER CACHES	PROB THAT 'STORE-BEHIND	ACK COUNTER* # 0	STORE CONTROLLER	WRITE INTO CACHE+UPDT DIREC	STCR UPDATES ITS DIREC	STCR SEARCHES & UPDATES	CACHE RECEIVES A CHANGE	STATUS ACK	CACHE UPDATES DIREC	CACHE DECR .STORE
VS1 JOB A4	XSMAXMP, 10	X\$PI-11,900	X\$PIN2,1000	XSDEX1,2	XSDEX2,4	X\$8EX1,1	X\$BEX8,8	X\$CBUS.1	XSREX, 4	XSKEX, 4	XSRPLR.8	XSRDEX1,6	X\$TIMER, 10000	X\$PRV1,1000			X\$PRV2.1000			X\$SHR, 1000					X\$CNT, 1000		XSSCS, 4	XSRDEX.8	XSCU.B	XSSCSU, 8	XSRDEX2,2		X\$REX1.8	XSDECR.4
: STCR	INITIÁL INITIÁL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL	INITIAL			INITIAL			INITIAL					INITIAL		INITIAL	INITIAL	INITIAL	INITIAL	INITIAL		INITIAL	INITIAL

SAME TOTAL TAN PROC. •
SUBA TOTAL EXEC TIMES •
SUBY TOTAL MAIT TIMES •
SUBY TOTAL ELAPSED TIME

VARIABLES

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

UMESP FVARIABLE (XSSUMT/XSNTXN) TXMT VARIABLE P3-P2

KEAN RESP TIME

PAGE 002

.

CONVERSATIONAL MONITOR SYSTEM \$\$AIR1,10/\$\$SIR1,10/\$\$TIR1,10/\$\$AIR1,10/\$\$OIR1,10 \$\$RIR2,10/\$\$SIR2,10/\$\$TIR2,10/\$\$AIR2,10/\$\$DIR2,10 S\$RID11,10/\$\$SID11,10/\$\$TID11,10/\$\$AID11,10 \$\$RID12,10/\$\$SID12,10/\$\$TID12,10/\$\$AID12,10 \$\$RID13,10/\$\$SID13,10/\$\$TID13,10/\$\$AID13,10 \$\$RID14,10/\$\$SID14,10/\$\$TID14,10/\$\$AID13,10 \$\$RID15,10/\$\$SID15,10/\$\$TID15,10/\$\$AID15,10 TXN MAIT TIME TXN EXEC TIME S\$41021,10/0\$S1021,10/S\$T1021,10 WICHA FUNCTION P1.05 1,4441/2,44412/3,4413/4,44415 1, MM411/2, MM412/3, MM213/4, MMM14/5, MMH15 STANA, 100, 100, 100 STANA, 100, 100, 100 VSTANA, 100, 100, 100 P3-P2-P4 7 PROC 23,19 STORAGE FOR DEVICES VS1 JOB STORAGE FOR L(1) CACHES STURAGE FOR RED STORAGE STORAGE VARIABLE Variable MICHE FUNCTION STDRA SE STORA GE STORA GE STORA GE STORA GE STORAGE TABLE TABLE TABLE FUNCTIONS FILE: STCR TABLES TXNA TXME TXMT

\*\*\*\*\*\*\*

\$\$RIK2,10/\$\$SIK2,10/\$\$IIK2,10/\$\$AIK2,10/\$\$DIK2,10 \$\$RQK2,10/\$\$SOK2,10/\$\$TOK2,19/\$\$AGK2,10/\$\$DOK2,10 \$\$q0K1,10/5\$50K1,10/5\$TIK1,10/5\$AIK1,10/5\$0UK1,10 FNUSLBUS1+SNFSDIR1+SNFSDOK1 FNUSGBUS+SNFSTIK! FNUSLBUSI-SNFSRIRI FNUSLBUSI-SNFSSIRI FNUSLBUSI-SNFSAIRI FNUSLBUSI-SNFSOIRI SNFSRIRI SNFSSIRI 44 \*\*\*\*\*\*\* \* \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \* \*\*\*\*\*\*\*\*\*\*\*\*\* STORAGE FOR K2, K3, K4 **VS1 JOB** READ-THADUGH BOOLEAN VARIABLES BVARIABLE BVARIABLE BVARIABLE BVARIABLE BVARIABLE 87221281E 87221281E 87221281E 87221281E 87221281E Bvariable Bvariable Bvariable Bvariable Bvari able Bvari able STORAGE FOR KI DRKJI BVARIABLE RTOKZ BVAZIABLE STORA GE STORAGE 8V F37 L(1) FILE: STCR BV FJR DRS1 DRA1 DRA1 DRA1 BROS1 RRS1 RRS13 RRS13 RRS13 BROS14 BROS14 BROS15 DRA9 DRA9 DRA9

CONVERSATIONAL MONITOR SYSTEM

FILE: STCR	STCR	VS1	VS1 JOB	**	CONVERSATIONA
DKS1	BVARI ABLE	m T	FNLS	FNUSLEUS1.SNrSSCK1	
	BVARI AB	ABLE	FNUSTBUST	9US1+SV-19UGA1	
K0111	BVARI AB	ABLE	FNUSL	FNUSLBUS1+S4+STID11	
KDT12	BVARI AB	ABLE	FRUSL	FNUSLEUS1*SNFSTID12	
KD113	BVAR I AB	ABLE	FNUS	FNUS_BUS1*5%F\$TZD13	
KDT14	BVARI AB	ABLE	Fisus:	FNUS_SUST-5N=STID14	
KDT15	<b>BVARIAB</b>	ABLE	1007	FACELEUS1-54-571015	
KDA11	BVARI AB	ABLE	FNUST	FNUSCISUST STATE TO 1	
KDA12	BVARIAB	ABLE	F. 3.5.	1,3051-54-54[012	
KDA13	BVARI AB	ABLE	FNUSLEUST	8024884888	
KDA14	<b>BVARIAB</b>	ABLE	FNUSL	54-54	
KDA15	BVARI AB	BLE	FNL\$L	SN = SA	
*****		•	*		
•				•	
* 8V FJR	IR INTER	LEVEL	EL COVA	• 5	
•				•	

FNU\$GBUS+SNESTK2 FNU\$GBUS+SNESSTK2 FNU\$GBUS+SNFSSCN2 FNU\$GBUS+SNFSTK1 FNU\$GBUS+SNFSTK1

KKR12 BVARIABLE KKS12 BVARIABLE KKO12 BVARIABLE KKT21 BVARIABLE KKA21 BVARIABLE

. BY FOR L(2) GPS

7	~	~	2	2	7	21	7	~	2	~	~	7	~
11	213	110	118	F\$2142	513	55102	Ë	A000	STOKE	ğ	ğ	80	Č
品化製は	in	151	1	ķ	11	i/i	j. 171	179	11	41.	Š		1/
3	S	.BUS2-5%	5	Š	NCS-EUSE-SN	Ś	FNUSLE SZ=S4=	Š	FNUSLBUSS+54	ŝ	US2*5N	ŝ	S
22	52	52	52	22	S	S	Š	EUS2-5	\$25	22	Š	U\$2*	us2.
190	FKU\$LBUS2-5	180	5	9	in or	9	נוו נוו	3	50.5	FNUSLBUS2+SI	J. H.	Δn	8
28	50	NUSLE	NUS	NUS.	53	3	33	FNUSL	S	S	FNUSL	NUSL	NUS! E
ĭ	Ĩ	Z.	£	'n.	Ē	Ľ	Z.	Y.	Z.	ĭ	Ľ	Z.	2
ш	щ	ш	Ę	w	щ	щ	щ	W	щ	щ	w	щ	w
A8	AB	ABL	AB	ABLE	ABLE	ABLE	A9LE	ABL	ABLE	ABL	ABLI	ABL	ABI
ARI ABL	ARI	18	ARI	ARI	IRI	BVARI	2	BVARI	BVARI	<b>H</b>	E	ARI	RI
2	2	2	8	8	BVAR	8	BVAR	8	2	BVAR 3	BVAR	8	2
8	7	24	A2	7	5	7	5	8	12	2	7	8	~
KRR.	<b>R</b> S2	KRT.	KRA	X S	RDR2	RDS2	RDT2	DKS2	X	A A	AKR2	<b>K</b> 02	Ž

\*\*\*\*\*\*\* MACROS

CONVERSATIONAL MONITOR SYSTEM

CONVERSATIONAL MONITOR SYSTEM

PAGE 006

TRANSIT TIME BY FOR SEND OP MACRO - SEND

STARTMACRO TEST E ENTER SE1ZE ACVANCE ASSIGN RELEASE LEAVE ENDMACRO SEND

\*\*\*\*\* MACRO - FINI

STARTMACRO MARK SAVEVALUE SAVEVALUE SAVEVALUE SAVEVALUE ASSIGN ASSIGN ASSIGN ASSIGN ASSIGN FINI

\* MACRO -USE

. #B USAGE TIME

STARTMACRO SEIZE ADVANCE ASSIGN RELEASE ENDMACRO

\*

VS1 JQB

FILE: STCR

CONVERSATIONAL MONITOR SYSTEM

FILE: STCR

	BEGIN	SIMULATION		• •	
	SIMULATE				
neo	1.4.0	•••	•	,	
*	RMULT	3,5,7,9,11,13,15,17	17		
CPU1 STAR1 RRE1	GENERATE PRIORITY MARK ASSIGN ADVANCE TRANSFER	SE S	SET HIGH P FOR NEW ARRIVAL TIME CPU ID	X X X	
DATA	IS IN DATA	CACHE			
RIN11 USE	ENTER MACRO Leave Advance	A1011 DE-11,X\$RDEX1 R1011 X\$CBUS	PUT TXN IN READ RE SEARCH AND READ CA FREE BUFFER	REQ BUFFER CACHE	
	TINI MANSERA ,ST	, STARI	A NEW TXN		
DATA	IS NOT	IN CACHE			
NIG11 USE SEND	ENTER MACRO PRIORITY MACRO	RIS11 S9211,XSREX 0 RID11,RIR1,LBUS1	PUT IN READ REQ BUFFER SEARCH DIRECTORY RESET PRIORITY ,XSBEX1, BVSDRR1	UFFER	
	TRANS FER	, COMR	TO COMMON CODE FOR	R READ	
WRITE	REQUEST	TO CACHE			

PUT TXN IN WRITE REQ BUFFER

S1011

WWW ENTER

SET HIGH P FOR NEW TXN ARRIVAL TIME CPU IO

X\$CBUS .X\$NREAD,WWW2,RRR2

ASSIGN ADVANCE TRANSFER

CONVERSATIONAL MONITOR SYSTEM

HIN12 FINI

,TRY2 SR1,1 SIR1

TRANSER ASSISN MACHO TPANSFER ASSIGN NTP12 USE SHR02 SEND

it.0 RRP1,X\$SCSU .X\$SHR,SHR12,SHR02 11.0 SIR1,SID12,LBUS1,X\$5EX1,3V\$RDS12

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                                                                                                                                                                                                                                                                                                                              PUT TXW IN READ REG BUFFER
SEAPCH AND READ CACHE
FREE BUFFER
                                                                                                                                                                                                                                                                                    SET HIGH P FOR NEW TXN
ARRIVAL TIME
CPU ID
                        DRP12,X$RDEX2
SID12.SOK1,LBUS1.X$BEX8,BV$DKS1
1.COMW
                                                                                                                                          1.SHR22
DRP12.X$RDEX2
SID12,SOK1,LBUS1.X$BEXB,BV$DKS1
1,COMW
                                                                                                                                                                                                                                                                                                                                                                                                                                        A NEW TXN
                                                                                                                                                                                                                                                                                                    1.3
X$CBUS
.X$NREAD,WWW3.RRR3
.X$PIN1,NIN13.RI%13
                                                                                                                                                                                                                                                                       ...XSMAXMP...F
                                                                                                                                                                                                                                                                                                                                                                                            AID13
DRP13, X$RDEX1
RID13
                                                                                                                                                                                  ,SfAR2
11,0
DRP13,X$REX1
SID13
                                                        ,STAR2
11.0
BV$D2D3,1
SID12
SID13
LBUS1
A$BEX1
4+,X$BEX1
LBUS1
 44
                                                                                                                                                                                                                                                             *******
                                                                                                                                                                                                                                                                                                                                                             . DATA IS IN DATA CACHE .
                                                                                                                                                                                                                                                                                                                                                                               ******************
                                                                                                                                                                                                                                                                                                                                                                                                                       XSCBUS
                                                                                                                                                                                                                                                                                                                                                                                                                                        STAR3
                                                                                                                                   SIR1
VS1 J08
             WACRO
AD WACRO
SPLIT
FINI WACRO
TANSFER
SHR12 ASSIGN
TEST E
FVIER
                                                                                                                                                                 SPLIT
WACRO
TRANSFER
ASSIGN
WACRO
                                                                                                                                                                                                                    ERWINATE
                                                                                                                                                                                                                                                                           GENERATE
PRIORITY
MARK
                                                                                                                                                                                                                                                                                                   ASSIGN
ADVANCE
TRANSFER
                                                                                                                                                                                                                                                                                                                                                                                                                                        FRANS FER
                                                                                 ENTER
SATER
ADVANCE
AESIGN
LEEVE SE
SPLIT
                                                                                                                                                                                                                                                                                                                                                                                                                       ADVANCE
                                                                                                                                                                                                                                                                                                                                                                                              ENTER
                                                                                                                                                                                                            LEAVE
                                                                                                                                                                                                                                                                                                                                                                                                                LEAVE
                                                                                                                                                                                                                                                                                                                                                                                                                               WACRO
FILE: STCR
                                                                                                                                                                                                                                            CP 7 #3
                                                                                                                                                                                          SHR22
USE
                                                                                                                                                                                                                                                                            CPV3
STA33
                                                                                                                                                                                                                                                                                                                                                                                              E-474
                                                                                                                                                  USE
SENO
                                                                                                                                                                          FINI
                                                                                                                                                                                                                                                                                                                             nung
nung
                                                                                                                                                                                                                                                                                                                                                                                                                               1814
```

. DATA IS NOT IN CACHE

CONVERSATIONAL MONITOR SYSTEM

VS1 JOB

FILE: STOR

FINI	MACRO		
	TRANS FER	STAR3	
SHR23	ASSIGN	11.0	
USE	MACRO	DRP14 XSREX1	
	LEAVE	51014	
	TERNINATE		
*****	*	*******	
•		*	
	CPU #4	•	
		# · · · · · · · · · · · · · · · · · · ·	
CPU4	GENERATE	XSWAXMPF	
STAR4	PRIORITY	6	SET HIGH P FOR NEW TXN
	MARK	2	ARRIVAL TIME
	ASSIGN	4,4	CPU ID
	ADVANCE	x\$cBUS	
RRR4	TRANSFER	.XSNREAD, WWW4, RRR4 .XSPIN1, NIN14, RIN14	24 414
*****	************	*****	
•	•	•	
. DATA	IS IN DATA	CACHE .	
•		•	
*****	***********	******	
AINIA	ENTED	41014	
USE	MACRO	DRP14 XSRDFX1	AND DEAD CACKE
 	LEAVE	81014	FREE BUFFER
	ADVANCE	SMUSX	
FINI	MACRO		
	TRANSFER	.STAR4	A NEW TXX
	•		
	化化苯甲基苯甲甲基苯甲基甲基甲基甲基甲基甲基甲基甲基甲基甲基甲基甲基甲基甲基甲基	*****	
. DATA	DATA IS NOT IN (	CACHE .	
•		•	
*****	**********	*******	
NINIA	ENTER	RID14	PUT IN READ RED BUFFER
SE	MACRO	DRP14, XSREX	RCH DIRECTORY
	PRICRITY	0	RESET PRIORITY
SEND	MACRO	RID14, RIR1, LBUS1	, X SBEX1, BV SDRR1
	TRANS FER	, COMR	TO COMMON CODE FOR READ
*****		• • • • • • • • • • • • • • • • • • • •	
•		•	
· WRITE	WRITE REQUEST TO CACHE	CACHE.	
	************	•	
4600	ENTER	51014	CANTAGE OF STREET IN MAT THE
USE	MACRO	DRP14, XSREX	WRITE DATA IN CACHE

CONVERSATIONAL MONITOR SYSTEM

PAGE 013

\*

FILE: STCR

NOT ISA

CONVERSATIONAL MONITOR SYSTEM

.XSPIN1,NIN15,RIN15

VS1 JOB

FILE: STCR

DATA IS IN DATA CACHE RRRS TRANSFER

PUT TXN IN READ BEQ BUFFER SEARCH AND READ CACHE FREE BUFFER A NEW TXN RID15 DRP15, XSRDEX1 RID15 XSCBUS STARS. 计分析分析分析 医多种性性坏疽 医拉拉斯氏性医检查检检 TRANSFER LEAVE ADVANCE WACKS RINIS ENTER USE WACRD FINI

DATA IS NOT IN CACHE

RID15 PUT IN PEAD RED BUFFER DRP15.X\$REX SEARCH DIRECTORY O RESET PRICRITY O RID15,RIR1,LBUS1,X\$BEX1,BY\$DRR1 NIN1S ENTER USE MACAD PAIDAITY SEND MACAD

TO COMMON CODE FOR READ COMR. TRANSFER

MITE REQUEST TO CACHE\*

PUT TXN IN MRITE REC BUFFER WRITE DATA IN CACHE DRP15, XSREX 51015 DECAN E3.43 zzz Zzz

SID15, SIR1, LBUS1, X\$BEX1, BV\$DRS1 YTI ACICS De34# SEND

RESET TAN PRIDRITY

11.0 .x\$PRV2.,NTP15 RRP1,X\$SCS SIR1 ASSIGN TRANSFER MACFO EASE 1875 SE

SR1.1 SIR1 .TRY5 11.0 RRP1.X\$SCSU .X\$SHR,SHR15.SHR05 11.0 5.FER 1EST E ENTER 1 TRANSFER NTP1S ASSIGN USE WECHO USE TRANSFER SMR05 ASSIGN

	MACRO		
SEND		SIR1, SID15, LECS1, ASBEAT, BVSKDS10	
USE	MACSO	DAP15, X\$RDEX2	
SEND	MACRO	SID15, SOA1, LEUS1, X SBEXB, BV\$DKS1	
	SPLIT	1.00%	
FINI	MACRO		
	<b>TRANS FER</b>	. 51425	
STRIS	ASSIGN	11.0	
		8/50501,1	
	ENTER	itais	
	FNTFD	TOTAL	
	SFIZE		
	ADVANCE	X	
	ASSIGN	1 (1 (1 (1 (1 (1 (1 (1 (1 (1 (1 (1 (1 (1	
	RELFASE	LEUS1	
	FAVE		
	50174	10 C C C C C C C C C C C C C C C C C C C	
3311	1111		
•	DE CAG	A CANADA A C	
SER	MACRO	SIOTO, SCAT, LILOT, ABGENG, BYBUAS!	
	SPLIT	1.00%	
FINI	MACRO		
		,STARS	
SHR25		11.0	
USE	MACRO	DRP11, XSREX1	
	LEAVE	SID11	
	TERMI NATE		
!			
8	COVINON CODE I	FOR READ REDUEST	
- 1	*******		
8	ASSIGN	11.0	
SE	MACRO	RRP1, x \$SCS	
	TRANSFER		
	FAVE	No.	
	TEST F	1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	
	ENTED	פוסל	
	TDANSFER	9700	
MTDV	ASSTON		
SEND	MACRO	RE91, ROK1, LB.51, XSBEX1, BVSRKR1	
		,	
USE	MACRO	KRP1, XSKEX	

ROK1, RIK2, GB.S. XSBEX1, BVSKKR12

MACRO MACRO MACRO

KRP2,XSKEX

RIK2, FIR2, LBJS2, XS BEX1, BVSKRR2

RRP2, XSREX

CONVERSATIONAL MONITOR SYSTEM

. READ DATA IS FOUND IN L(2)

VS1 JOB

FILE: STCR

RRR21 ASSIGN

- DATA IS IN D21

RIR2, RID21, L 3US2, X \$BEX1, BV\$RDR21 MACRO

SEND

USE

DRP21, XSDEX2 MACRO

RID21, TOK2, LBUS2, X \$BEX8, BV\$DKT2 MACRO

SEND

\*\*\*\*\*\*\*\*\*\*\*\*\*\* . READ-THROUGH TO L(1) \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

KRP2, XSKEX MACRO

USE

TOK2, TIK1, GBUS, XSBEXB, BV\$RTOK2 MACRO

SEND

STORE DATA INTO L(1) AS RESULT OF A READ-THROUGH

11.0 STOR1 ASSIGN

KRP1,XSKEA MACRO

USE

1, FNSWICHE SPLIT TGANINATE

\* RT STORE INTO D11

11,0 WWHI ASSIGN

\*

TIM1, TID11, LBUS1, X \$BEX8, BV\$KOT11 MALRO SEND

FILE: STCR	STCR	VS1 JUB A4	CONVERSATIONAL MONITOR SYSTEM
USE	MACRO	DRP11, XSRPLR	
	SPLIT	1.0VL11	
	TEST E	BVSDRKD1,1	
	ENTER	0181	
	ENTER	00K1	
	SEIZE	LBUS1	
	ADVANCE	X\$BEX1	,
	ASSIGN	4+, XS3EX1	
	RELEASE	LBUS1	
	LEAVE	11011	
	SPLIT	1,00/1	
USE	MACRO	RRP1, XSCU	
	LEAVE	0131	
	TERMI FATE		
CVL11	-	11 0	
	ADVANCE	XSCBUS	
FINI	MACRO		
	TRANS FER	R ,STAR1	
****	*********		
•		• •	
, X	SIUKE INIU DIZ		
	*******		
KEE 77	WWW12 ASSIGN	11.0	
SEND	MACRO	TIK1, TID12, L3US1, X \$8EX8, BV\$KDT12	X SBEXE, BV\$KDT12
USE	MACRO	DRP12, XSRPLR	

SPLIT 1,0VL12
TEST E BV\$DRK01,1
ENTER 01R1
SEIZE LBUS1
ADVANCE X58EX1
ASSIGN 4+,X8EX1
RELEASE LBUS1
LEAVE TID12
SPLIT 1,0VL1
USE MACRO RRP1,X\$SCU
LEAVE 01R1
TERMINATE
DVL12 ASSIGN 11,0
ADVANCE X5CBUS
FINI MACRO
TRANSFFR ,STAR2
\*\*RT STORE INTO D13 \*\*

```
CONVERSATIONAL MONITOR SYSTEM
                                       TIK1, TID13, LBUS1, X $8EX8, BV$KDT13
                                                        DRP13, XSRPLR
                                                                                                             LBUS1
X$BEX1
4+,X$BEX1
LBUS1
TID13
1,0VL1
RRP1,X$SCU
                                                                               1.0VL13
BVSD2KO1.1
 4
                                                                                                                                                                                       11.0
x$CBUS
                                                                                                                                                                                                                                                       TAANSFER STAR3
                        11.0
VS1 JOB
                                                                                                                                                                                                                                     RT STORE INTO D14
                                                                                                                                                                               ETMI NATE
                                                                                                                                                                                      DVL13 ASSIGN
ADVANCE
FINI WACRO
                                                                                                                                                                                                              TRANS FER
                                                                                                ENTER
ENTER
SELVE
LOVANCE
LSSIGN
                        WWW13 ASSIGN
                                                                                                                                                                        EAVE
                                        MACRO
                                                         MACRO
FILE: STCR
                                        SEND
                                                         USE
```

PAGE O18

TID14 1.0vL1 RRP1,X\$SCU 0IK. 1.0VL:4 BY\$DRKG1,1 0IR1 CGK1 LBUS1 X\$BEX1 4+.X\$BEX1 LBUS1 ENTER ENTER SEIZE ASTANCE ASSIGN RELEASE WACRO USE

TIK1, TID14, LBUS1, X\$BEX8, BV\$KDT14

11.0

BEST4 ASSIGN

DRP14, XSRPLR

WACRO

WACRO

SEND

XSCBUS ADVANCE WACRO TAANS FER

F142

**ERMINATE** 

EAVE

**Tesign** 

2170

\*

CONVERSATIONAL MONITOR SYSTEM

VS1 JCB

FILE: STCR

TIN1.TID15.LBUS1.X\$3EX8,8V\$KDT15

WWW15 ASSIGN

MACRO

SEND

\* RT STORE INTO D15

DR215, K\$42\_R MACRO

1.04-75 BV\$29401.1

TID:5 1.0/01 RRP1.45SCU OIR1 +. XSEEX1 11.0 x\$C3.5 K\$554 ERMI NATE ENTER SEIZE ADVANCE ASSIGN RELEASE ADVANCE OVL15 ASSIGN USE

\*\*\*\*\*\*\*\*\*\*\*\*\* TRANSFER STATE . HANDLE OVF FROM L(1)

FIHI

ASSIGN 27.0

DOK1. DIK2, GBUS. XSBEX1, BVSKKD12 KRP1.XS4EX MACRO MACRO

SEMO USE

USE

KRP2,XSAEX

DIKZ.CIRZ.LBUSZ.X\$ BEX1, BV\$KR02 MACRO MACRO

0182 LEAVE TERMINATE

RRP2.XSREX

MACRO

USE

SEND

. COMMON CODE FOR STURE-BEHIND

COMP	ASSIGN	11,0
USE	MACRO	KRP1.X\$KEX
SEND	MACRO	SOM1.SIM2.GBUS,XSBEXB,BV3KKS12
USE	MACRO	KRP2, XSKEX
SEND	MACRO	SIK2,SIR2,LBUS2,X\$BEX8,BV\$KRS2
USE	MACRO	rrp2, xsrex
SEND	MACRO	SIR2, SID21, LBUS2, X\$BEX8, 8V\$RD521
USE	MACRO	ORP21, X\$DEX2
SEND	MACRO	S1021, AOK2, LBUS2, X\$BEX1, BV\$DKS2
¥Ç.	EG 2.	L(2) TO L(1)
ACK21	1 ASSIGN	11.0
USE	MACRO	KRP2, XSKEX
SEND	MACRO	AOM2, AIK1, GBUS, XSBEX1, BVSKKA21
USE	MACRO	ERD1. XSKEX
	SPLIT TERMINATE	1, FNSWICHA
*****	************	***
ACK	ACK HANDLED BY	D11 .
	*****	• • • • • • • • • • • • • • • • • • • •
AAATT	NSISS I	11,0
SEND	MACRO	AIK1, AID11, LBUS1, XSBEX1, BVSKDA11
USE	MACRO TRANS FER	PRP11,XSDECR .XSCNT,,CNf11
CHTTT	LEAVE TERNIMATE ( ASSIGN	AID11 11.0 AID11, AIR1, LBUS1, X\$BEX1, BV\$DRA1

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                                                                                                                                          ATM1, AID15, LBUS1, X$BEX1, BV$KDA15
                                                                                        11.0
AID14,AIR1,LBUS1,X$BEX1,BV$DRA1
RRP1.X15CU
AIR1
                                                                                                                                                                                                                                                                         11.0
AID15.AIR1.LBUS1,X$BEX1,BV$DRA1
RRP1,X$SCU
AIR1
                                  DRP14, XSDECR . XSCNT, CNT14
                                                                                                                                                                                                                            DRP15, XSDECR
A4
                                                             A1014
                                                                                                                                                                                                                                                        A1015
VS1 J0B
                                                                                                                                                                                                                                                                                                                                               * SIMULATION CONTROL
                                                                                                                                                      . ACK HANDLED BY D15
                                                                                                                                                                                                                                                       LEAVE
TERMINATE
CNT15 ASSIGN
SEND WACRD
USE MACRD
                                   MACRO
TRANSFER
                                                                                                                                                                                                                             MACRO
TRANS FER
                                                                                                                                                                                                                                                                                                                     TERMINATE
                                                                                                                                                                                         AAA15 ASSIGN
                                                              LEAVE
                                                                                                                                                                                                           MACRO
FILE: STCR
                                                                                                                                                                                                          SEND
                                   USE
                                                                                                                                                                                                                            USE
```

GENERATE TERMINATE START END

APPENDIX (III): The "CC/PTB/C" Program

//C EKEC PGW=DAGO1,TIME=&TLIMIT
//C EKEC PGW=DAGO1,TIME=&TLIMIT
//STEPLIB DD DSN=PGTLUCK.LIBRARY.GPSS.LGAD.DISP=SHR
//DINTPUT DD DSN=PGTLUCK.LIBRARY.GPSS.LGABLKSIZE=931
//DINTERO DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=1383
//DSNMTAB DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=7:12
//DREPIGEN DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=3030
//DINT#GDR DD UNIT=SCRATCH.SPACE=(CYL.(1,1)).DCB=BLKSIZE=2680 REALLDCATE FUN.6.QUE,10.FAC.50.BVR,200.BLD,2000.VAR.50 REALLDCATE FSV.50.HSV.10.COM.39968 // PEND //STEP1 EXEC GPSS.PARM=C,TLIMIT=9 //DINPUT1 DD \* //\*PASSWORD SCUBA //GPSS PROC

CC/PTB/C

TXN PARM USAGE

CPU ID
TXN ARRIVAL TIME\*
TXN COMPL TIME \*
TXN EXEC TIME \*

MODEL COMPONENTS

BUSES: GEUS. LBUS1...
CACHES: D11...D15
LEVEL CONTR.: 41, K2
REQ PROCS: E2
DEVICES: D21
STORAGE: 81. RO
STORAGE: 51. SO
STORAGE: 11. TO
STORAGE: A1. AO
STORAGE: A1. AO

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

4

VS1 JOB

FILE: CCC

//TAR1 JOB TAR, // PROFILE='DEFER' // TIME=5

CONVERSATIONAL MONITOR SYSTEM

INITIAL

INITIAL INITIAL

INITIAL INITIAL INITIAL INITIAL

FILE: CCC

INITIAL INITIAL INITIAL INITIAL INITIAL

NIXM TOTAL TXM PAGG. \*
SUMM TOTAL EXEC TIVES \*
SUMM TOTAL MAIT TIVES \*
SUMT TOTAL ELAPSED TIW\* \* SAVEVALUES

MEAN RESP TIME TXN ELAPSED TIME TXN WAIT TIME TXN EXEC TIME (XSSUMT/XSNTXN) P3-P2 P3-P2-P4 F4 MRESP FVARIABLE TXNT VARIABLE TXNM VARIABLE TXNX VARIABLE

TABLES

VSTX\1,100,100,100 VSTX\4,100,100,100 VSTX\X,100,100,100 TABLE TABLE TABLE TXNT TXNU

**FUNCTIONS** 

\*

MCMST FUNCTION P1,05

CONVERSATIONAL MONITOR SYSTEM

LOGKUP PLUS READ/WRITE TIME OF D11 SIKULATION TIME (IN 10 NS UNITS) TIME R1 SEARCHS DIREC & UPDT LRU STATUS

INITIAL INITIAL INITIAL

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* VARIABLES

```
CONVERSATIONAL MONITOR SYSTEM
```

VS1 J08

FILE: CCC

1,STAR1/2,STAR2/3,STAR3/4,STAR4/5,STAR5

MICHW FUNCTION PS.DS 1.WWW11/2.WWW12/3.dww15

MUNIT FUNCTION PS.DS 1,WDD11/2,MDD12/3,WDD13/4,WDD14/5,WDD15

RUNIT FUNCTION PS.05 1,RDD11/2,RDD12/3,RDD13/4,RDD14/5,RDD15

LOAD FUNCTION R41,05

WICHA FUNCTION PS.DS 1. AAA 1/2. AAA 12/3. AAA 13/4. AAA 14/5. AAA 15

STORAGE FOR L(1)

\*

\$\$RID11,10/\$\$SID11,10/\$\$TID11,10/\$\$AID11,10 \$\$RID12,10/\$\$SID12,10/\$\$TID12,10/\$\$AID12.10 \$\$RID13,10/\$\$SID13,10/\$\$TID13,10/\$\$AID13.10 \$\$RID14,10/\$\$SID14,10/\$\$TID14,10/\$\$AID14.10 \$\$RID15,10/\$\$SID15,10/\$\$TID15,10/\$\$AID15.10 STORAGE STORAGE STORAGE STORAGE STORAGE

STORAGE FOR DEVICES

\*\*\*\*\*\*\*

S\$RID21,10/S\$SID21,10/S\$TID21,10 STORAGE

STORAGE FOR REG PROC

\*

\$\$RIR1,10/5\$SIR1,10/5\$TIR1,10/5\$AIR1,10/5\$OIR1,10 \$\$RIR2,10/5\$SIR2,10/5\$TIR2,10/5\$AIR2,10/5\$OIR2.30 STORAGE STORAGE

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* STORAGE FOR KI \$\$ROK1,10/5\$SOK1,10/5\$TIK1,10/5\$AIK1,10/5\$OCK1,10 STORAGE

**VS1 JOB** 

FILE: CCC

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

. STORAGE FOR K2, K3, K4

\$\$RIK2,10/5\$5IK2.:0/5\$TIK2,10/5\$AIK2,10/5\$0IK2,10 \$\$RDK2,10/5\$5DK2.16/5\$TDK2,10/5\$ADK2,10/55DDK2,10 STORAGE STORAGE

BOCLEAN VARIABLES

READ-THROUGH BV FOR FNUSGBUS+SNF STIK1 STOKE BVAPIABLE

BY FOR L(1)

FNUSLBUST-SN-SSIST FNUSLEUS1 = SN F S1231 BVARIABLE BVARIABLE BVARI ABLE BVARI ABLE BVAR J ABLE BVARI ABLE **BVARIABLE** BVARI ABLE **BVARIABLE** BJARI ABLE BVAZI ABLE BYARI ABLE BVARI ABLE CBR1

FNUSLBUS1+SNFSTID14 FNUSLBUS1+SNFSTID15 FNUSLBUS1+SNFSAID11 FNUSLBUS1+SNFSAID12 FNUSLBUS1+SNFSAID13 FNUSLBUS1+SNFSAID14 FNUSLBUS1+SNFSAID15 LBUS1\*SNF\$TID:1 LBUS1\*SNF\$TID:2 LBUS1\*SNF\$TID:3 FNUSLBUS1+SNFSROK1 FNUSLBUS1+SNFSSCK1 FNUSLBUS1+SNFSCCK1 FNUSE FNUSE BVARI ABLE CAUCA ACTION OF ACTION OF

FNUSGBUS\*SNF\$9LKZ FNUSGBUS\*SNF\$SIKZ FNUSGBUS\*SNF\$SOKZ FNUSGBUS\*SNF\$TIK1 FNUSGBUS\*SNF\$TIK1 KKR12 BVARIABLE KKS12 BVARIABLE KKO12 BVARIABLE KKT21 BVARIABLE KKA21 BVARIABLE BY FOR L(2) OPS

FNUSLBUSZ\*SNFSRIRZ FNUSLBUSZ\*SNFSSIRZ FNUSLBUSZ\*SNFSTIRZ FNUSLBUSZ\*SNFSRIRZ FNUSLBUSZ\*SNFSRIDZ FNUSLBUSZ\*SNFSRIDZ FNUSLBUSZ\*SNFSTIDZ FNUSLBUSZ\*SNFSTIDZ FNUSLBUSZ\*SNFSTIDZ FNUSLBUSZ\*SNFSTICKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ FNUSLBUSZ\*SNFSACKZ EVARIABLE BVARIABLE KRS2 KRT2 KRT2 KRA2 KRA2 RRD21 RD521 RD721 DKS2 DKT2 DKT2 BKR2 RKA2

MACRUS

USAGE TIME MACRO -USE #A FACILITY =

STARTMACRO SEIZE ADVANCE ASSIGN RELEASE ENDMACRO

\* BY FOR INTER LEVEL COM\*

VS1 JOB

FILE: CCC

BEGIN SIMULATION

		•	
0 4		6 4 4 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6	
TEST E ENTER SELTE ADVANCE ASSIGN RELEASE LEAVE ENDMACRO	ONS - 08	STARTMACRO TEST E ENTER SEIZE SEIZE ASSIGN RELEASE ENDMACRO	10 - FINI
	EMCRO	95	• MACRO

3 NTXN+,1 SUMX+,V\$TXNX SUMH-,V\$TXNW SUMT+,V\$TXNT ARESP,V\$MRESP 2.0 3.0 STARTMACRO MARK SAVEVALUE SAVEVALUE SAVEVALUE SAVEVALUE ASSIGN ASSIGN ASSIGN ASSIGN FINI

VS1 JOB

FILE: CCC

MACRO - SEND

/E.1 STARTMACRO TEST E SEND

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                                                                                                                                                                                                                                                                                                          SET HIGH P FOR NEW TAR
ARRIVAL TIME
1.4
CPU ID
.XSNREAD.WWW1.READ
                                                                                                                                                                                                                                                                                     SET HIGH P FOR NEW TXN
ARRIVAL TIME
1,3
CPU ID
,X$NREAD,WWW1,READ
                                                                                                                                                                                               SET HIGH P FOR NEW TXN
ARRIVAL TIME
1.2
CPU ID
.XSNREAD,WWW1.READ
                                                                                               SET HIGH P FOR NEW TXN
ARRIVAL TIME
1,1
CPU ID
,X$NREAD,WWW1.READ
                                                                                                                                                                                                                                                                                      *******************
     4
                                                                                    ************************
                                                SIMULATE
      VS1 JOB
                                                                                                                                                                                                                                                                                                                                                                                           GEMERATE
PRIORITY
MARK
ASSIGN
TRANSFER
                                                                                                                                                                                                                                                                                                    GENERATE
PRIORITY
WARK
                                                                                                                                                                                                                                                                                                                            ASSIGN
TRANSFER
                                                                                                                                                                                                              CPUZ GENERATE
STARZ PRIORITY
MARK
                                                                                                               GENERATE
PRIORITY
MARK
                                                                                                                                                                                                                                           TRANS FER
                                                                                                                                       ASSIGN
TRANSFER
                                                                                                                                                                                                                                     ASSIGN
                                                                                                                                                                                                                                                                                                                                                                CP. 14
                                                                                                                                                                                                                                                                          C$ 0.43
                                                                                                                                                                                   CPU #2
                                                                      CPU #1
       FILE: CCC
                                                                                                                                                                                                                                                                                                       CPU3
STAR3
                                                                                                                   CPU1
STAR1
```

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

CPU #5

**VS: JOB** 

FILE: CCC

SET HIGH P FOR NEW IXI.
ARRIVAL TIME
1.5
CPU ID
.XSNREAD,WWW1.READ generate Priority Mark

CPUS STARS

ASSIGN TRANS FER

DATA TO BE READ

assign assign trans fer

READ

DEVICE D11

5.1 RID11, LBUS1, X\$BEX1, BV\$RDR11 .X\$PIN1, NIN11, RIN11 11.0 DRP11, X\$RDEX1 LBUS1 ASSIGN MACRO TRANSFER ASSIGN MACRO SEIZE \$500 \$860

XSBEX1 4+,XSBEX1 IBUS1 RIN11 USE

1. FNSWCHST ADVANCE ASSIGN RELEASE LEAVE MACRO

DRP11,X\$REX RID11,ROK1,LBUS1,X38EX1,UV\$DKR1 ,COMR SPLIT TERMINATE ASSIGN PRIORITY MACRO MACRO NIN1 USE SEND

• DEVICE D12

CONVERSATIONAL JUNITOR SYSTEM

VS1 J09

FILE: CCC

DRP12, XSREX RID12, RDK1, LBUS1, X\$BEX1, BV\$DKR1 ,COMR 5,2 RID12, LBUS1, XSBEX1, BV\$ROR12, X\$PIN1,NIN12,RIN12,11,0 1 . FNSWCHST KSBEX1 4+, KSBEX1 LBUS1 RID12 11.0 MACHO SEIZE ADVANCE K\$BE? ASSIGN 4+,R RELEASE LBU' LEAVE RIT LEAVE RIT TEMINATE TEMINATE 1 TEMINATE 1 TEMINATE 1 TEMINATE 1 TEMINATE 1 MACRO TRANS FER RDD12 ASSIGN SND WACRO TRANSFER RIN12 ASSIGN USE WACRO USE SEND

. DEVICE D13

RIDIS, LBUSI, XSBEXI, BVSRDRIS, XSPINI, NINIS, RINIS, 11.0 DRPIS, XSRDEXI TRANSFER TRANSFER USE PACRO RDD13 ASSIGN SND MACRO

I. FNSWCHST XSBEX1 4-.XSBEX1 LBUS1 RID13 SPLIT TERMINATE 11113

DRP13, XSREX RID13, ROK1, LBUS1, XSBEX1, 8'\$DKR1 .COMP 11.0 MACRO MACRO TRANS FER MINID ASSIGN PRIORITY

\*\*\*\*\*\*\*\*\*\*\*\*\*\* DEVICE D14

MDD14 ASSIGN

3

VS1 J0B

FILE: CCC

DRP14,XSREX RED14,ROK1,LBUS1,X\$BEX1,BV\$DKR1 ,COMR RIDIA, LBUSI, XSBEXI, BVSRDRIA , XSPINI, NINIA, RINIA 11,0 DRP1A, XSRDEXI 1 . FNSHCHST K\$BEX1 4+,X\$BEX1 LBUS1 RID14 11.0 ERMINATE PRICRITY WACRO WACRO TRANSFER RANS FER ADVANCE ASSIGN PELEASE ASSIGN WACRO SEIZE ASSIGN #1#14 USE MIN14 USE SEND FIMI 8

DEVICE DIS

5.5 RID15, LBUS1, XSBEX1, BVSRDR15 .XSPIN1, MIM15, RIN15 11.0 DRP15, XSRDEX1 LBUS1 SND MACRO TRANSFER NIM1S ASSIGN USE MACRO RDD15 ASSIGN

X\$BEX1 4+,X\$BEX1 LBUS1 R1015 ADVANCE ASSIGN RELEASE LEAVE

1. FNSWCHST **ERWINATE** FINI

11.0 PRIORITY ASSIGN NINIS

DRP15,XSREX RID15,ROK1,LBUS1,X\$BEX1,BV\$DKR1 ,COMR WACRO MACRO TRANS FER use seno

. DATA TO BE WRITTEN IS IN LEVEL 1

WWW ASSIGN

CONVERSATIONAL MONITOR SYSTEM

VS1 JOB

FILE: CCC

ASSIGN TRANSFER

\* DEVICE D11

5.1 \$1011, LBUS1, X\$BEX1, BV\$RDS11 DRP11, X\$RDEX1 ASSIGN MACRO PRIDRITY MACRO SPLIT YACRO SPLIT TERMINATE

1 FNSWCHST

DEVICE 312

5.2 SID12, LBUS1, X\$BEX1, 8V\$RDS12 DRP12, X\$RDEX! MD012 SMD USE

1. FNSWCHST

FIMI

DEVICE 213

SID13, SOK1, LBUS1, X \$BEX8, BV\$DK51, COM4 SI013, LBUS1, X\$BEX1, BV\$R0513 0RP13, X\$R0EX1 FACRO FACRO FACRO FACRO SNO SNO SNO

1 FNSWCHST SPLIT MACRO SPLIT ERRINATE

**SEND** FIMI

7

S, FNSLOAD FNSHUNIT

SID11, SDK1, LBUS1, X \$BEX8, BV\$DK511, COMM

SEND FINI

SID12, SOK1, LBU51, X\$BEX8, BV\$DK51, 1, COMM #4CPO PRIOR ITY PRIOR ITY SPLIT ELCRO

TERE : NATE

CONVERSATIONAL MONITGE SYSTEM

7 FILE: CCC

VS1 JOB

DEVICE 3:4

SID14, SOK1, LBUS1, XSBEX8, BVSDKS1 5.4 SID14, LBUS1, XSBEX1, BVSRD514 DRP!4, XSRDEX1 TTI MCIR

1 . FNSWCHST

FINI

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

SI015, SOK1, LBUS1, X\$BEX8, BVSDKS1

ROK1, RIK2, GBUS, X\$BEX1, 3VSKKR12 MACRO

SEND

RIK2.RIR2, LBUS2, XS BEX1, BVSKAR2 MACRO SEND

MACRO

USE

DEVICE D15

5.5 SID15.LBUS1, X\$BEX1,BV\$23515 DRP15,X\$RDEX1

SPLIT WACRO SPLIT TERMINATE

1, FNSHCHST

CO-MICH CODE FOR READ REQUEST

ASSIGN

XRP1, XSKEX MACRO

KRP2, XSKEX MACRO

JSE 1

RRP2,X\$REX

READ DATA IS FOUND IN L(2)

DATA IS IN D21

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

11,0 RRR21 ASSIGN

RIRZ, RID21, LBUS2, X\$BEX1, BV\$RDR21 MACRO

SEND

RID21, TOK2, LBUS2, X \$ BEXB, BV \$ DKT2 DRP21, XSDEX2 MACRO

MACRO

SEND

USE

\* READ-THROUGH TO L(1)

\*\*\*\*\*

KRP2,XSKEX MACRO

USE

TOK2, TIK1, GBUS, X\$BEX8, BV\$RTOK2 MACRO \* STORE DATA INTO L(1) AS RESULT OF A READ-THROUGH

11,0 STOR1 ASSIGN KRP1,XSKEX MACRO 1. FNSWICHW SPLIT TERMINATE

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

. RT STORE INTO D11

CONVERSATIONAL MONITOR SYSTEM

44

VS1 J0B

FILE: CCC

7

```
TIK1, TID11, LBUS1, X $BEX8, BV$KDT11
                                             TID11.00K1, LBUS1, X$BEX1, BV$DK01
                            DAPIL, KIRPLR
                                                                        1, FNSACHST
                                                                                                               .0711
                                                                                                                              RT STORE INTO 012
                                                                            SPLIT
TERMINATE
                                                                                                 OVF11 TRANSFER
WWW11 ASSIGN
                                                     MACRO
SFLIT
MACRO
                               MACRO
                MACRO
                SEND
                                                      SEND
                                                                    FINI
                                 USE
```

TIM1, TID12, LBUS1, X \$ E E X8, B V \$ KD T 12 TID12,00K1,LBUS1,X\$BEX1,BV\$DKG1 1,0VF12 DAP12, XSRPLR 1, FNSWCHST WWW12 ASSIGN MACRO MACRO MACRO SPLIT SEND FINI SEND

.0711 OVF12 TRANSFER

RT STORE INTO D13

TIK1, TID13, LBUS1, XSBEXB, BVSKDT13 WWW13 ASSIGN SEND

TID13,00K1,LBUS1,X\$BEX1,BV\$DK01 1,0VF13 MACRO SPLIT MACRO SPLIT TERMI NATE SEND FINI

FILE: CCC

VS1 JOB

11.0

MACRO SPLIT TERMINATE

MACPO

DRP13,X\$RPLR MACRO

1, FNSWCHST

.00. OVF13 TRANSFER

RT STORE INTO D14

WWW14 ASSIGN

TIM1, TID14, LBUS1, X \$BEX8, SV\$KDT14 SEND MACRO

DRP14. XSRPLR MACRO

USE

TID14,00K1, LBUS1, X \$8EX1, BV\$DK011,0VF14

SEND

FINI

1, FNSWCHST

MACRO SPLIT MACRO SPLIT TERMINATE

.0VL1 OVF14 TRANSFER

\*\*\*\* \* RT STORE INTO D15

\*\*\*\*\*\*

TIK1, TID15, LBUS1, X \$BEX8, BV\$KDT15 WWW15 ASSIGN MACRO

SEND

USE

DRP15.XSRPLR MACRO

1. FNSWCHST MACRO SPLIT MACRO SPLIT TERMINATE

TID15,00K1,LBUS1,X\$BEX1,BV\$DKQ11,0VF15

SEND

FINI

OVF15 TRANSFEP

\*\*\*\*\*\*\*

\* HANDLE OVF FROM L(1)

11.0 ASSIGN 2

KRP1,XSKEX MACRO

FILE: CCC

FILE: (	, ,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,	VS1 JCB	A4 CONV	CONVERSATIONAL MONITOR SYSTEM
SEND	MACRO	99X1	0941,3142,GBUS,X\$BEX1,BV\$KKD12	
USE	MACRO	KRP2	HRP2, FSMEX	
SEND	MACRO	0142	DI42.3IR2, LBUS2, X\$ BEX1, BV\$KR02	
USE	MACRO	2008	8402.4 <b>\$</b> REX	
	LEAVE TERMINATE	01#2		
8	COMYON CODE FOR		STORE-BEHIND	
<b>4</b> 00	ASSIGN	11,0	ι)	
USE	MACRO	K OP 1	REP1. XSKEX	
SEND	MACRO	SCK1	SCK1.SIK2,GBUS,X\$BEX8,BVSKKS12	
USE	MACRO	X	KAP2.FSKEX	
SEND	MACRO	SIK	SIK2, SIR2, LBUS2, X\$ BEX8, BV\$KRS2	
USE	MACRO	RRS	RRP2.XSREX	
SEND	MACRO	)E1S	SIG2.SID21,LBUS2,X\$BEX8,BV\$RDS21	.21
USE	MACRO	089	DRP21, XSDEX2	
SEND	MACRO	SID	SID21, AGK2, LBUS2, X\$BEX1, BV\$DKS2	ć.
ACK	( FROM L(2) TO L(1)	2) TO L	(1)	
USE	MACRO	KRP	MRP2,X\$KEX	•

1.FNSWICHA

ACK2, AIK1, GBUS, XSBEX1, BVSKKA21

KAP1.XSKEX

MACAG

USE

SEND MACRO

SPLIT TERMINATE

11.0  AIK1, AID11, LEUS1, XSBEX1, BV\$KDA11  DRP11, X\$REX  AID11  AIK1, AID12, LBUS1, X\$BEX1, BV\$KDA12  DRP12, X\$REX  AID12  AIK1, AID13, LBUS1, X\$BEX1, BV\$KDA13  DRP13, X\$REX  AID13  AID13  AID13
---

CONVERSATIONAL MONITOR SYSTEM

VS1 JOB A4

FILE: CCC

DAP14, XSREX

MACRO

USE

AID14 LEAVE TERMINATE

. ACK HANDLED BY D15

11.0 AAA15 ASSIGN

AIK1, AID15. LBUS1, X \$8EX1, 8V\$KDA 15 SEND MACHO

AIDIE LEALE TERMINATE

DRP15. XSREX

CECYA

USE

SINULATION CONTROL

XSTIMER GENERATE TERRINATE START END

SS EMD

APPENDIX (IV): The "CC/PTB/P" Program

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                  REALL DCATE FUN.6, QUE, 10, FAC, 50, BVR, 200, BLD, 2000, VAR, 50
REALL DCATE FSV, 50, HSV, 10, COM, 39968
                                                                                                                                                                                                                    EXEC GPSS, PARM=C, TLIMIT=9
                                                                               TIME TO BE SELECT STREET
 4
                                                                                                                                                                                                                               TXN ARTVAL TIME IXN CCUPL TIME IXN EXEC TIME
                                                                                                                                                                                                                                                                                                                                                                  ********************
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      · 经保存股份 医安特特氏征 经过的公司的 医动物性 1
                                                                                                                                                                                                                                                                                                                                                                                                          BUSES: GSUS. LBUS1...
,51 JOB
                                                                                                                                                                                                                                                                                                                                                                                                                   CACHES: D11....D15
LEVEL CONTAL: 41.K2
                                                                                                                                                                                                                                                                                                                                                                                     MODEL COMPONENTS
                             /TAR1 JOB TAR.
/ PROFILE='CEFER'.
/ TIME=5
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  MODEL PARANETERS
                                                            /*PASS#040 SCJBA
/GPSS PROC
                                                                                                                                                                                                                                                   TXN PARM USAGE
                                                                                                                                                                 //STEP1 EJ
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                STORAGE
STORAGE
STORAGE
FILE: CCP
                                                                                                                                                         CN3d //
                              //TAR1
```

NTXN TOTAL TXN PROC. \*
SUMX TOTAL EXEC TIMES \*
SUMM TOTAL WAIT TIMES \*
SUMT TOTAL ELAPSED TIM\* \*\*\*\*\*\* \* VARIABLES (X\$SUMT/X\$NTXN) P3-P2 P3-P2-P4 FVARI ABLE VARIA BLE VARIA BLE VARIA BLE MRESP TXNT TXNM TXNM

TXN ELAPSED TIME
TXN MAIT TIME
TXN EXEC TIME MEAN RESP TIVE

TABLES

VSTXNT,100,100,100 VSTXNW,100,100,100 VSTXNX,100,100,100 TABLE TABLE TABLE TXNX TXNT

SAVEVALUES

.......... FUNCT I DNS 2.5 MCHST FUNCTION

PAGE 002

FILE: CCP

```
CONVERSATIONAL MONITOR SYSTEM
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         $$$181,10/$$51R1,10/$$11R1,10/$$A1R1,10/$501R1,10
$$R1R2,10/$$51R2,10/$$11R2,10/$$A1R2,10/$$01R2,10
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 S$RCK1,10/S$SCK1,10/S$TIK1,10/S$AIK1,10/SSCCK1,10
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         $$\text{S$\text{S$\text{S$\text{S$\text{S$\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\text{S}\tex
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        S$41021.10/S$SID21,10/S$TID21,10
                                                                                                                                                                                                                                                                                                    MICHW FUNCTION P5.25
|.WWW11/2.WWW12/3.EAN3/4.MWW14/5.WWW15
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           WUNIT FUNCTION 25,05
1,WDD11/2,WDD12/3,40013.4,WDD14/5,WDD15
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         RUNIT FUNCTION 75.75
1,RD011/2,RD012/3,RCC13 4,RCD14/5,RD015
                                                                                                                                                                                  1,STAR1/2,STAR2/3,STAP3/4,STAR4/5,STAR5
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              ********************
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  LOAD FUNCTION RN1,05
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    STORAGE FOR K2.K3,K4
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     . STORAGE FOR REQ PASC
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                . STORAGE FOR DEVICES
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          STORAGE FOR L(1)
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             **************
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  STORAGE FOR KI
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         STORAGE
STORAGE
STORAGE
STORAGE
STORAGE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        STORAGE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         STORAGE
STORAGE
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      STORAGE
FILE: CCP
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           CACHES
```

CONVERSATIONAL MONITOR SYSTEM VS1 J09

\$\$RIK2.10/SESIK2.10/S\$TIK2.10/S\$AIK2.10/5\$OIK2.10 \$\$ROK2.10/S\$SOK2.10/S\$TOK2.10/S\$AOK2.10/5\$OOK2.10 STORAGE STORAGE

\*\*\*\*\*\* BOOLEAN VARIABLES

BV FOR READ-THROUGH

FNUSGBUS+SNFSTIK1 RTOK2 BYARIABLE

BV FOR L(1)

FNUSLBUSI+SN FSRID11 FNUSLBUSI+SN FSRID12 FNUSLBUSI+SN FSRID13 FNUSLBUSI+SN FSRID14 FNUSLBUSI+SN FSRID15 FNUSLBUSI+SN FSSID11 FNUSLBUSI+SN FSSID12 FNUSLBUSI+SN FSSID13 FNUSLBUSI+SN FSSID13 FNUSLBUSI+SN FSSID14 FNUSLBUSI+SN FSSID15 FNUSLBUSI+SN FSSID15 FNUSLBUSI+SN FSSID15 FNUSLBUSI+SN FSSID15 FNUSLBUS1+SNFSRIR1 Byariable Byariable Byariable By ARIABLE By ARIABLE BLARIASLE BVARI ABLE BLARI ABLE BYAPI ABLE SVARI ABLE B:42I ABLE BYA21 ABLI E. L. I ABLI 

FNU\$LBUS1+SNF\$20K1 FNU\$LBUS1+SNF\$50K1 FNUSLBUS1+SNFSGOK1 BYSHI ABLE BYARI ABL

FNUSLBUST+SNFSAIDT3 FNUSLBUST+SNFSAIDT4 FNUSLBUST+SNFSAIDT5 FNUSLBUS1\*SNF\$TID1 1 FNUSLBUS1\*SNF\$TID1 2 FNUSLBUS1\*SNF\$TID1 3 FNUSLBUS1+SNFSAID115 FNUSLBUS1+SNFSAID11 FNUSLBUS1+SNFSAID12 FNUSLBUS1+SNF\$T1D14 BVARI ABLE BVARI ABLE BYARI ABLI WAPIABLE STAPIABLE BVARI ABLI E. AP! ABLI B. 141 ABLI BVARI ABL BY 184 18F DKR1
DKS1
DKO1
KDT11
KDT12
KDT12
KDT13
KDT15
KDT15
KDA12
KDA13
KDA14

BY FOR INTER LEVEL COM.

\*

BYARI ABLE

PAGE 004

FILE: CCP

FILE: CCP

PAGE 005

FN.SLBUS2-SN-581R2 FN.SLBUS2-SN-5S1R2 FN.SLBUS2-SN-5S1R2 FN.SLBUS2-SN-5S1R2 FN.SLBUS2-SN-5S1R2 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D21 FN.SLBUS2-SN-5S1D22 FVJSGBUS+SNFSRIKZ FVJSGBUS+SNFSSIKZ FVJSGBUS+SNFSSKZ FVJSGBUS+SNFSTIKT FVJSGBUS+SNFSTIKT KKR12 BVARIABLE KKS12 BVARIABLE KKO12 BVARIABLE KKT21 BVARIABLE KKA21 BVARIABLE BY FOR L(2) CPS BVARIABLE BVARIABLE BVARIABLE BVARIABLE BVARI ABLE **BVARI ABLE** BVARI ABLE KRR52 KRR52 KRR22 KRR22 KRR22 RD521 BDK22 BKR22 RKR22

MACROS

MACRO -USE

Ę

SEIZE ADVANCE ASSIGN RELEASE ENDMACRO

MACRO - SEND

USAGE TIME

STARTMACRO SE

BEGIN SIMULATION

VS1 JOB

FILE: CCP

3.5,7.9,11,13,15,17 CPU1 STAR1

CPU #1

SET HIGH P FOR NEW TXN ARRIVAL TIME 1.1 CPU ID .XSNREAD, WHW1, READ Generate Paicrit Wark ASSIGN TRANSFER

CPJ #2

SET HIGH P FOR NEW TAN ARRIVAL TIME 1.2 CPU ID .XSNREAD, WWW1, READ STAR2 CONCASTE STAR2 ENCASTY WARA AS5184

CPJ 03

TRANSFER

SET HIGH P FOR NEW TXN
ARRIVAL TIME
1.3
CPU IO
.K\$NREAD.WWW1.READ CPU3 GEYEALTE STAR3 PRICATTY BARK assign Trans Fer

CPJ #4

SET HIGH P FOR NEW TXN ARRIVAL TIME 1,4 CPU ID .x\$nread,www1,Read GENERATE PRIOZITY WARK ASSIGN TRANS FER STAR4

CONVERSATIONAL MONITOP SYSTEM 4

VS1 JOB

FILE: CCP

\*\*\*\*\*\*\*

CPU #5

SET HIGH P FOR NEW TXN
ARRIVAL TIME
1,5
CPU ID
.XSNREAD,WWW1,READ GENERATE PRIORITY MARK

CPUS STARS

ASSIGN TRANS FER

DATA TO BE READ

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

医脊髓管 计存储设计 经经济保险 计自动设计 计设计设计 计电影电话 医电影性医学

11.0 5,FN\$LOAD FN\$RUNIT ASSIGN ASSIGN TRANS PER

READ

\* DEVICE 011

RDD11 ASSIGN

5.1 ,x\$PIN1,NIN11,RIN11 11,0 RID11,LBUS1,X\$BEX1,BV\$RDR11 DRD11,X\$DEX1 LBUS1 X\$BEX1 4+,X\$BEX1 LBUS1 RIN11 ASSIGN SND MACRO USE WACRO

RID11 ADVANCE ASSIGN PELEASE

1. FNSWCHST MACRO SPLIT TERMINATE ASSIGN FRIORITY MACRO TRANS FER NIN11 FINI

ROK1, LBUS1, XSBEX1, BV\$DKR1, COMR 11.0

SXB

\*\*\*\*\*\* . CEVICE D12

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

5.2 RDD12 ASSIGN

44

ASSIGN PELEASE

.FNSACHST

TERMINATE ASSIGN PRICRITY MACRO TRANSFER

AD4", LBUS1, XSBEX1, BVSDKR1.

DEVICE D13

RDD13 ASSIGN

. X52 L 41, NIN1 3, RIN1 3

RCAT, LBUS1, XSBEX1, BVSDKR1, CCMR

DEVICE D14

.XSPIN1,NIN14,RIN14 11.0 RID14,LBUS1,XSBEX1,BV\$RDR14 DRP:1,XSDEX1

VS1 JC5

FILE: CCP

.4571N1,NIN12,RIN12 11.6 RID:2.LBUS1,XSBEX1,BV\$RDR12 DF=12,XSDEX1 LE.51 XSSE41 44.15EEX1 LECS:

MACRG

RIN12 SND USE

3213S

FINI

NIN12

SND

PICTA, LBUST, XSBEXT, BVSRDR13 TRANSFER RIN13 ASSIGN SND MACRO

XSSE(1 4+,XSSEX1 LBC51 RID13 ADVANCE ASSIGN RELEASE LEAVE MACRO

I.FLEACHST FIRI

0,1 TERMINATE ASSIGN PRICA ITY MACRO TRANS FER EININ SNO

RDD14 ASSIGN TRANS FER RIN14 ASSIGN SND MACRO USE MACRO

Ą

VS1 JOB

FILE: CCP

ROK1, LBUS1, X SBEX1, BV\$DKR1, COMR X\$BEX1 4+,X\$BEX1 LBUS1 RID14 1.FNSKCHST 0.10 NIN14 ASSIGN PRIORITY SND MACRO TRANSFER ERMI NATE ADVANCE ASSIGN RELEASE LEAVE MACRO SPLIT FINI

DEVICE D15

11,0 RID15, LDUS1, X\$BEX1, BV\$RDR15 DRP15, X\$DEX! LBUS1 ROK1, LBUS1, X SBEX1, BV SDKR1, COMR 5.5 .xspin1,nin15.Rin15 1, FNSWCHST X\$9EX1 4+.X\$BEX1 LBUS1 RID15 11.0 RDD15 ASSIGN
TRANSFER
RIN15 ASSIGN
SND MACRO
USE MACRO
USE ADVANCE
ADVANCE
ASSIGN
RELEASE
LEAVE
FINI MACRO
SPLIT ERMI NATE ASSIGN PRIORITY MACRO TRANSFER NIN15

DATA TO BE WRITTEN IS IN LEVEL 1

SAC

S,FRSLOAD,FNSWUNIT ASSIGN TRANS FER ASSIGN

\*\*\*\*\*\*\*\*\*\*\*\*\*\*

DEVICE D11

5,1 SID11, LBUS1, X\$BEX1, BV\$RDS11 DAP11, X\$DEX1 MASSIGN WACGO WACGO WACGO SPLIT WACGO SPLIT TERMINATE MOOT 1 SNO USE

SID11, SOK1, LBUS1, X \$BEX8, B\SDKS1, 1, COMW

SEMO FINI

1 . FNSWCHST

DEVICE 312

5,2 SID12, LBUS1, X\$BEX1,BV\$RDS12 DRP12,X\$DEX1 0 MACRO MACRO PRICRITY WACRO MOO12

SID12, SOK1, LBUS1, XSBEX8, BVSDKS11, CDMW 1.FNSWCHST

FINI SEVO

SPLINE MACHINE SPLINE TEMENATE

DEVICE DE

5.3 SID13, LBUS1, X\$3EX1, BV\$RDS13 DRP13, X\$DEX1 ASSIGN MACRO WACRO PRICRO SND N

SID13, SOK1, LBUS1, X SBEX8, BYSDKS11, COMM 1.FNSWCHST SPLIT WACRO SPLIT TERMINATE

FINI

CHES

DEVICE D14

VS1 JOB FILE: CCP

44

VS1 JOB

FILE: CCP

SID14.SOK1, LBUS1, X \$BEX8, BV\$DKS1 1,COMW 5.4 SID14, LBUS1, X\$BEX1, BV\$RDS14 DRP14, X\$DEX1 0 1.FNSHCHST WDD14 ASSIGN
SND MACRO
USE MACRO
PRIORITY
SEND MACRO
SPLIT
FINI MACRO
SPLIT
TERMINATE

DEVICE D15

\*

\*\*\*\*\*\*\*

SID15, SOK1, LBUS1, X SFEX8, BV\$DKS11, COMM 5,5 SID15,LBUS1,X\$BEX1,BV\$RDS15 DRP15,X\$DEX1 0 1.FNS#CHST MACRO PRIORITY MACRO SPLIT MACRO SPLIT ASSIGN NDD15 / SND 1 USE 1 SEND FINI

COMMON CODE FOR READ REQUEST

KRP1, XSKEK 11.0 ASSIGN MACRO

KRP2, XSKEX MACRO

USE

MACRO

SEND

ROK1.RIK2,GBUS,X\$BEX1,BV\$KKR12

RIKZ, RIRZ, LBUSZ, XS BEX1, BVSKRRZ RRP2,XSREX MACRO MACRO SEND USE

READ DATA IS FOUND IN L(2)

\* DATA 15 14 22"

\*\*\*\*\*\*\*\*\*\*\*\*\*

=1=2. aID21, L BUS2, X \$8EX1, 3V\$RDR21 RRR21 455131

3==2. . KSDEX2 CK DYM SACAD

SENO

=132. TOK2, LBUS2, X \$BEX8, BV\$DKT2 SEND KASAS

\* READ-THROUGH TO LITE 44=2, X SKEX CFUVA TEKE, BYSRINGBUS, XSBEXB, BYSRIDKE 0F04A

. 491 . XSKEX CECYA.

USE

RT STORE INTO DIT

0.1 WWW11 ASSIGN

TIA1, FID11, LBUS1, X SBEX8, BVSKDT11 SEND MACRO

CAPIT. XSDEXI MACRO USE

7

FILE: CCP

SEND USE

. STORE DATA INTO LIT) AS RESULT OF A READ-THROUGH

STORI ASSIGN

1.FNSHICHE SPLIT TERMI 4ATE

FILE: CCP		VS1 JQB A4	CONVERSATIONAL	MONITOR SYSTEM
SEND SP FINI 44	VACRO SPLIT VACRO SPLIT TERMINATE	TID11,00K1,LBUS1,X\$BEX1,BV\$DK01 1,0VF11 1,FM\$&CHST	<b>5</b> 0x0 1	
0VF11 TR	TRANS FER	, 04 L1		
* RT STORE	RE INTO DI	~		
•	*****	• • • • • • • • • • • • • • • • • • • •		
WWW12 AS	ASSIGN	11.0		
AW CM32	WACRO	TIK1, TID12, LBUS1, XSBEX8, BYSKD712	SKD712	
USE	WACRO	DRP12, XSDEX1		
SEND SP SP SP SP SP SP SP SP SP SP SP SP SP	WACRO SPLIT WACRO SPLIT TERMINATE	T1D12.00K1,LBU51,X\$BEX1,BV\$DK01 1.0VF12 1.FN\$WCHST	\$ DK 0 1	
DVF:2 14	TRANS FER	יסארו.		
a 47 STORE	. z	TO D13		
heal3 AS	NSISSN	11.0		
M CN3S	WACRO	TIK1, TID13, LBUS1, X\$BEX8, 5,5KDT1	<b>5</b> KD713	
350	WACRO	DRP13, X\$DEX1		
SEA3 43	VACAO SPLIT VACRO SPLIT TERMINATE	TID13.00K1,LBUS1,XSBEX:,B.5DKG1 1.0VF13 :,FNSWCHST	10703	
CVF13 TR	TRANSFER	.001		
RT STORE	RE INTO 01	• • •		
		•		

FILE: S	150 633	VS1JOB A4 CONVERSATIONAL MONITOR SYS	R SYS
	•	e : : : : : : : : : : : : : : : : : : :	
WHETA.	NC1551	11.0	
SEND	CeCYA	TIK1,TID14,LE: -(,X\$BEX8,BV\$KDT14	
USE	CEC 7.	DRP14,XSDEX1	
SEND	# W # W # W # W # W # W # W # W # W # W	TID14, COK1, Lt. 7 X \$3EX1, BVSDKG1	
0VF14	-24%5 -ER	.סענו	
•			
• RT ST	S:0=E :NTO 015	* * *	
		***************************************	
MW15	VE::SST	11,0	
SEND	26018	TIM1, TID15, LBUS1, X \$5EX8, BV\$MDT15	
USE	<b>V</b> ±030	DAP15.X\$DEX1	
SEND	WACAO SPLIT WACAO SPLIT ERVINATE	TID15,00K1,LBUS1,X\$BEX1,BV\$DKO1 1,0VF15 1,FNSWCHST	
0VF15	+3+85+	יסארו.	
	***********		
HAAD	100 F 100 E		
	ASS:34	0	
USE	VAC23	KRP1.XSKEX	
SEND	CECTA	OOK1,GIK2,GBUS,X\$BEX1,EV\$KKO12	
USE	06044	KRP2,X\$KEX	
SENO	BACHO	DIK2.DIR2,LBUS2,X\$BEX1,BV\$KRO2	
USE	DECTA	ARP2,X\$REX	

CONVERSATIONAL MONITOR SYSTEM 4

VS1 JOB

FILE: CCP

LEAVE DIRZ TERMINATE

COMMON CODE FOR STORE-BEHIND

11.0 ASSIGN

KRP1,XSKEX MACRO

SOK1, SIK2, GBUS, XSBEX8, BVSKKS12 KACRO

SEND

KRP2.XSKEX WACRO

SIKZ, SIRZ, LBUSZ, XS BEXB, BVSKRSZ MACRO

RRP2,XSREX WACRO

SEND

USE

USE

SIR2, SID21, L9US2, X SBEXB, BVSRDS21 **VACRO** 

DRP21, XSDEX2 WACRO

SEND

USE

SID21, AOK2, LEUS2, XSBEX1, BVSDKS2 WACRO

SEND

• ACK F92M L(2) TO L(1)

ADK2, AIK1, GBUS, X\$BEX1, BV\$KKA21 KRP2,XSKEX WACRO WACRO

KRP1, KSKEX AIK1 WACRO LEAVE

SEND

USE

USE

-ERMINATE

SIELLATION CONTROL

.5" JCS A4

FILE: CCP

GENERATE TERNIKATE START END

ASTINER

SS END